Exploiting Implicit User Activity for Media Recommendation

Michele Trevisiol

TESI DOCTORAL UPF / 2014

Directores de la tesi

Prof. Dr. Ricardo Baeza-Yates
Department of Information and Communication Technologies

Dr. Alejandro Jaimes Larrarte
Yahoo Labs

Acknowledgments

This PhD thesis has been a long and amazing journey...

I discovered the world of research while doing my Master Thesis between University of Padua and Yahoo! Research Barcelona, and I will be forever grateful to Massimo Melucci, Ricardo Baeza-Yates and Roelof van Zwol for making this path possible. But much before, everything started with my mum and dad, that have always pushed me up as much as I could (and put me back together every time I fell), and with my sister that is always there, no matter how far we are, she is a star that I always see enlighted.

In that very early research step, I learned a lot from one of the best "crack" that I ever met, Lluis. I had the opportunity to work with people that have become more than colleagues, such as Francesco, Börkur, Yannis, Marco Serafini, Andrea. Our paths have changed now, but I know that I will always be able to count on them. I was very lucky to have met the unusual side of their amazing personality (I will always remember the "francescata").

My journey moved forward when I started the PhD thesis with Alejandro Jaimes, and found Luca Chiarandini, a fellow traveller that has stayed by my side through thick and thin. He is another crack that it is not easy to find, and I had the pleasure to work and to learn with him, in many many occasions. I'd like to thank Janette, the best soul with whom you can share the positive side of life, always with a smile and ready to embrace the life how it comes; I must also mention her desk, often invaded since it was empty most of the time. I am also grateful to Ruth and Eduardo, that shared with me the beautiful pain of this trip. Many other friends and researchers have helped me in this travel: first of all, Luca Aiello, my ideal mentor that

has helped me to face many obstacles, by trusting and supporting me and, with its endurance he still continues to amaze me every night we go out. Gianmarco, a friend; one of those friends that are there, and that you do not want to be apart. With him I've shared a long path of this journey and I am grateful that our ways have crossed for so long so far. Cigdem, a dark beautiful flower that I will never forget. It was just a terrible mistake that I have started to **hike the rings** so late, but now I am sure I will not move back.

I had the luck to have great colleagues, such as Nicola that is always ready to help you with advices or to push you for a night into a disco pub; Paloma that will always succeed to tear you a laugh and eat all my preferred stocks of dark chocolate. Rossano and Francesco, with whom I would definitely like to share much more time. Ioanna, who has recently started her PhD, roller coasting between stress, fear... but also with satisfaction and big beautiful smiles. I am only sorry of not being there when her journey is more pleasant. Daniele, one of the best researchers and craziest persons that I've met; I like the way he looks for what he wants, I would have liked to learn more from him. Ana, a lovely and unexpected surprise that I met in the last months of my journey. Maybe without knowing it, but she made me see things from different perspectives.

I will never forget the lab, where I spent most of my time in the last years, and its two souls, who have made it a beautiful, cheerful, and funny place: Natalia and Estefania. We have invested so much time in extremely important gossip meetings and I had also the pleasure to help them at organizing, celebrating, and enjoying certain events at Yahoo.

The lab has been a great place to work and learn, with many other persons like Vanessa, Ulf, Adam, Neil, Roi, Jordi, Puya, Ilaria, Peter...

And finally Miriam, one of the reasons why it was so nice going to the lab everyday, and maybe the main reason why it will be so difficult to move forward.

Apologize for many others that I have surely forgot to write down here. I've tried to be concise and to focus on the people that have been more related to my PhD. I would have many other words to spend for amazing persons like Ale and Herve', and a long long list of other people that I will always keep in my mind.

Thanks.

Abstract

This thesis explores in depth how to use the user browsing behavior, and in particular the referrer URL, in order to understand the interest of the users. The aim is, first, to understand the preferences of the users from their navigation patterns, *i.e.*, from the implicit actions of the users. Then, to exploit this information to personalize the content offered by the service provider. The key findings from our studies allowed us to propose different solutions in terms of recommender systems and ranking approaches for media items. We show how the browsing behavior of the users captured by the browsing logs is extremely meaningful to understand new users and to estimate their preferences.

Resumen

Esta tesis analiza de modo exhaustivo el comportamiento del usuario en la web y, en particular, su interacción con las URLs recomendadas, para así conocer sus intereses. El objetivo fundamental es, en primer lugar, entender las preferencias de usuario a partir de sus patrones de navegación por la web, estudiando sus acciones implícitas. En segundo lugar, se trata de aprovechar esta información para personalizar el contenido ofrecido por el proveedor de servicios. El resultado de estos estudios nos ha permitido proponer diferentes soluciones en términos de sistemas recomendadores y ranking de productos multimedia. De este modo, hemos podido demostrar cómo el comportamiento del usuario en la web, obtenido a partir de registros de navegación, es extremadamente útil para comprender a nuevos usuarios y poder así estimar sus preferencias.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, social media platforms have grown very fast in terms of number of users involved. As a result, the data and the information that are spread through these platforms, and that can be found online, have grown exponentially. Nowadays, web users have access to more information and more resources than they ever had. Social media platforms are updated continuously, photo- or video-sharing websites have billions of media items available, news portals have thousands of new articles per day, and the list could cover almost any type of website. Users need to explore a large amount of information in a limited period of time, often losing themselves in information overload. As a consequence, users struggle to find what they are looking for, and often they could miss interesting information. In the context of specific domains like news portals or social media sites typically rich of data, the information overload is a fundamental problem that needs to be properly addressed. These represent the most visited domains in the Web, since the online newspapers cover almost 40% of the overall Internet visits, and social media websites around 20%,[1] and the volume has considerably increased in the last years (as reported by Nielsen [92, 19]).

In order to help and guide the users into the huge collection of data, their interests and preferences need to be understood. Nevertheless, users are rarely motivated[2] [111] to give feedback to the service provider even if this leads to an improvement of their experience. Users with this behavior are

[1] http://www.people-press.org/2012/09/27/section-2-online-and-digital-news-2/

[2] http://blog.elatable.com/2006/02/creators-synthesizers-and-consumers.html

known as *passive*. On the opposite, *active*[3] users on a social network, are those users that are actively contributing to the website. For example, by making actions (*e.g.*, liking, sharing, commenting), by managing contents, or by expanding their social connections. These signals provided by this type of users, are meaningful to understand their taste and their preferences. In Flickr, it was observed [93] that active users account for approximately 10% of the Flickr population. This percentage is in line with many other social platforms and websites [111]. As a consequence, it becomes really difficult to understand what kind of photos the majority of the users prefer.

The user's feedback can be classified into two types: *explicit* and *implicit*. The first one refers to the set of *actions* that the user does, directly specifying his or her personal opinion in relation to a context. A well-known example is Netflix, where users votes on a stars-scale how much they enjoyed the movie or series. In social networks like Facebook or Twitter, some of the actions made by the users clearly indicate their level of interest, *i.e. likes*, *retweets* or *favorites*. In general, users can express their opinion, rate or share the content, or make other actions that are likely to indicate their interest. In e-commerce platforms like Amazon and eBay, *purchases* are another indicator of user taste. In those cases, the service provider can collect these actions for each user and build a *user profile*, *i.e.*, a set of user characteristics and preferences, with the aim of improving its service for the specific user.

Explicit preferences usually represent clean and reliable data that might contain both, positive and negative opinions. Implicit actions, instead, are related to the users opinions but they are automatically inferred from the users' behavior. Some examples are the sequences of pages that the user visits, the time spent on each page, the clicks performed inside the page, the URL from where the user is accessing the page, and so on. On one hand, this different data is very noisy, since the users' preferences have to be *interpreted*, and they are not directly declared by the users themselves. On the other hand, the implicit actions are always available since they are extracted from the navigation logs, unlike explicit actions that imply the user to be active on giving his feedback. This is one of the reasons that this data is an extremely powerful source of information.

In some cases, the users are unknown, that is, there is no information about their previous visits, and the only knowledge available about them is represented by the implicit actions they are currently doing in the website (*e.g.*, web pages that they are visiting, time spent on each page, the actions that

[3] http://www.thinkingit.com.au/blog/definition-of-active-users

they are performing, *etc.*). These cases are known as *cold-start* problems, and these types of users are often called *newcomers* or *new visitors*. Any action the user performs on a website is stored into logs that keep track of all the web pages the user visited, together with additional data the user provides. These logs are generated at the server-side and they contain information such as the browser used, the location, the previous page visited by the user (called *referrer URL*), the timestamp, and so on. Most of this information is very helpful to gain some insight about the user, especially in the context of newcomers where it might be the only information available. Collecting and analyzing previous navigation logs allows to identify users with similar behavior.

This thesis focuses on how to exploit the implicit information extracted by the users navigation patterns, that is, the web pages visited by the users. In particular we consider the *newcomers*, tackling the cold-start problem. We propose different solutions and we discover novel approaches to personalize the information content at the user level.

We show how to exploit this information, focusing particularly in the referrer URL. We tackle this problem by using the *BrowseGraph*, a graph built on the user navigation sessions where the nodes are the web pages and the edges are the transitions. The *BrowseGraph* was previously used for computing *page importance* [72] revealing to be more reliable and dynamic than the standard hyperlink graph. This thesis introduces the *BrowseGraph* in the context of news and photo-sharing websites, with the goal of improving ranking and recommendation of media items such as articles and photos or videos. We define the notion of domain-dependent *BrowseGraph* that we call *ReferrerGraph*, namely a graph composed by the browsing sessions of users coming from the same referrer domain (*e.g.*, users coming from *facebook.com*). Our studies and experiments show how these graphs can exploit the *referrer URL*, retrieving information about the newcomers and improving services such as recommendations. This reveals to be extremely important for the service provider, as the results reported in this work allow to personalize the navigation of the users and increase their engagement finding interesting content.

The structure of this thesis is synthesized in Figure 1.1.

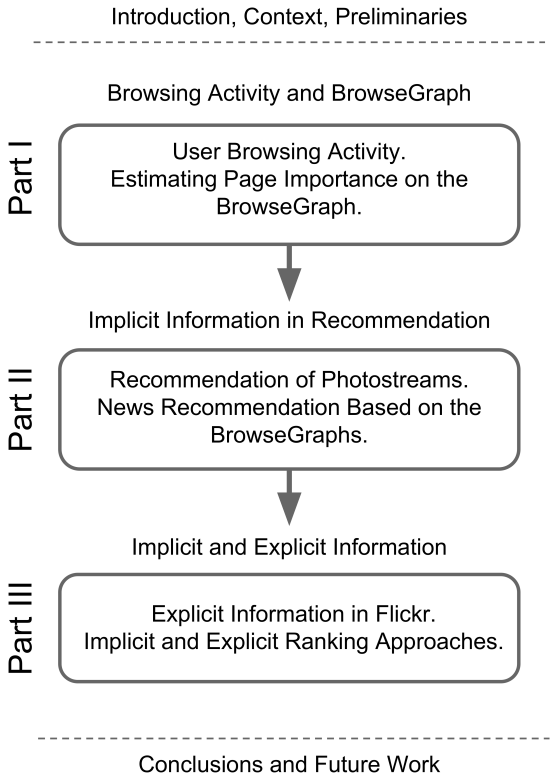


Figure 1.1: The figure summarizes the contents of the dissertation, with the three main parts that contain the results, preceded by an introductory and contextual discussion and followed by the general conclusions.

1.1. Research Problems

In this section we summarize the research questions and the contributions of this thesis.

As mentioned, the vast majority of users do not explicitly give any feedback about their taste and preferences. This lack of information for the service provider, translates into the impossibility to promote personalized content of high interest to the users. Thus, the service provider has to extract meaningful information from implicit user feedback.

Q1. *How could the content of interest be promoted to the users if we do not know anything about them? Is it possible to perform a personalization of content to newcomers?*

This problem is known as *cold-start*, and in this context, it refers to how to deal with unknown users[4]. When there is little user explicit information, the cold-start problem has been addressed extending the user profile using, for example, association rules [95, 91, 66]. Other solutions are based on mapping models, where the implicit user feedback are mapped into explicit ratings and then an explicit-based algorithm is used [82, 83, 6, 25, 53]. However, these approaches require a minimum set of implicit information about the user and, in some cases, also the explicit feedback in order to build the mapping model. In our case, we focus on a cold-start problem where the user enters for the first time inside the system and we do not have any previous information, neither implicit nor explicit.

Any action that a user performs on a website is stored into logs that keep track of all the web pages visited with a set of additional data that the user provides. This information is always available at the server side. In this thesis we focus on mining this source of data exploiting meaningful information in order to understand the user, as we show starting from Chapter 4. We exploit the *BrowseGraphs* since, due to its dynamic characteristic, it has been proof [72] to be more reliable to capture the users interests and behavior. In Chapters 5 and 7 we introduce a new structure called *ReferrerGraph*, that is a sub-graph computed by the browsing sessions but it is strictly dependent of the external referrer, namely the last URL or domain visited by the user before entering into the network. This graph collects the behavior of the users that come from the same external domain (*e.g.*, Facebook or Yahoo). We use them to model the user behavior and predict what the user is going to visit and what kind of sessions she will perform. To our knowledge, these graphs, and the approaches applied, were never used before for any of the presented applications.

The *BrowseGraph* has been used before only as alternative source for computing the importance of web pages. Anyhow, this graph differs by definition from the well-known hyperlink graph and due to its limited diffusion it has not been well studied in literature. However, we have not guarantee that

[4] *Cold-start* refers to the issue that the system cannot draw any inferences for users or items when there is not enough information. In this thesis we consider only the users' cold-start problem.

computing PageRank-like measures on this (sub-)network leads to reliable results since this graph has been studied very little before.

Q2. *How do PageRank-like algorithms behave on the BrowseGraph and on the ReferrerGraphs? Are these graphs reliable in this context to understand user behavior?*

The *BrowseGraph* has been compared to the standard hyperlink graph [72, 73] using different centrality-based algorithms. However, there is a lack of research about browsing graphs due to its novelty. One of the few works was presented by Lee *et al.* [124] where they made a similar comparison building a web spam identifier. Their experiments showed that algorithms performed on the browsing graph outperform those on the original graph. In this thesis we perform an in-depth analysis on the reliability of the users' browsing graphs, with centrality-based techniques such as PageRank [81], in order to support their use.

In Chapter 5 we perform experiments of random walk-like algorithms on *BrowseGraphs* and *ReferrerGraphs*, to understand the limits and the advantages of them. Our goal is to study and analyze the applicability of these types of algorithms on this graph. One of the problem that we tackle is the well known *Local Ranking Problem*, widely studied for the hyperlink graph (see [29, 37, 7, 18, 17]), but never considered in the browsing graph. This is related to the computation of centrality-like algorithms on a *local* graph where the scores of the nodes, and the ranking itself, could significantly differ with respect to the one computed on a *global* graph where all the nodes and edges are known. We analyze what are the differences between the local computation of PageRank with respect to the global graph when we expand step-by-step the local graph with its neighbors. Our studies are the first of this kind on any graph based on browsing sessions. Among other findings, we found that expanding the *BrowseGraph* with few neighbors with high importance leads to a faster convergence. This outcome will support the approach used on this thesis, in particular in Chapter 7.

The information about the users navigation patterns included in the *BrowseGraph* allows to extract meaningful information about the behavior of the users. Moreover, the *ReferrerGraphs* help to characterize the users by distinguishing different interests and types of web pages visited.

Q3. *Would it be possible to exploit these browsing graphs in order to recommend novel and interesting content to users? What is the contribution of these graphs compared to standard methods based on implicit or explicit data?*

In this thesis, after showing how it is possible to predict the actions of the users with the *BrowseGraph* and the proposed *ReferrerGraphs*, we perform various ranking and recommendation tasks using different sources. We briefly show, in Chapter 8, how the user makes explicit actions in a photo-sharing platform such as Flickr, and how the mechanisms of a social platform (*e.g.*, interacting with other users, sharing contents) increase the engagement and the users interest. Finally, in Chapter 9, we compare standard approaches based on the users' implicit and explicit feedback and we discuss the difference between such applications and the random walk approaches based on the *BrowseGraph*. While in the literature such comparison was not explored, our proposed evaluation metrics are effective to understand the difference of the methods applied.

1.2. Structure and Contributions

This thesis begins with three chapters that describe the overall context of the dissertation, then three main parts describe the results, and finally a chapter discusses the main conclusions.

Context and Methodology

These two chapters describe the context of this thesis, the problems to deal with, and the general approaches and datasets used.

Chapter 2: A description of the work that has been presented in the state of the art, highlighting the difference and the contribution of the thesis is presented.

Chapter 3: This chapter describes the datasets and different set of algorithms and approaches that are used in this thesis.

Part I. Browsing Activity and BrowseGraph

The second part of the thesis introduces a practical case of user browsing behavior. It shows how to mine the web log data, how to model the user sessions, and how to exploit the referrer URL. The goal is to understand the behavior of the user by his or her navigational patterns. In addition, two fundamental structures of this thesis are introduced and explained: the *BrowseGraph* and the *ReferrerGraph*, that are analyzed deeply in Chapter 5.

Chapter 4: A first study on the user browsing behavior inside the Flickr website network is presented. It exploits an important source of information: the *external referrer URL*, namely the last URL visited by the user before entering in the current website. The study shows the rich informativeness of the referrer URL, and how to exploit it in order to identify the user behavior. The content of this chapter is a result of collaboration with Luca Chiarandini, co-author of the following article:

- [32] Luca Chiarandini, Michele Trevisiol, and Alejandro Jaimes, "Discovering Social Photo Navigation Patterns", *IEEE International Conference on Multimedia and Expo (ICME 2012)*, pp. 31-36, Melbourne, Australia, July, 2012.

Chapter 5: This chapter describes a deeper analysis on different *ReferrerGraphs*, namely a browsing graph built on sessions with the same referrer URL. The behavior of users coming from different URLs is analyzed on Yahoo News data: different referrer URLs often reflect different behaviors. Moreover, a study in time of the evolution of the *BrowseGraphs* is performed in conjunction with the computation of random walk like algorithms and their performance. A well-known problem, called Local Ranking Problem is tackled, and two different applications are proposed. This chapter reinforces the importance of the *BrowseGraphs*, a structure that is vastly used on the experiments of this thesis, and it proves how random walk-like algorithms are meaningful also on this type of graphs. The content of this chapter is based on the following article:

- Michele Trevisiol, Paolo Boldi, Luca Maria Aiello and Roi Blanco, "Local Ranking Problem on the BrowseGraph", *under review.*

Part II. Implicit Information in Recommendation

The third part of the thesis implements different recommender systems based on implicit data. Two main applications are built and evaluated, the first one on Flickr recommending entire set of photos, and the second one on Yahoo News recommending articles. Both the applications present strong challenges in the context of recommending items from huge and highly sparse collections (*i.e.*, Flickr and Yahoo News) where it is fundamental to drive the user towards the most interesting content.

Chapter 6: A Flickr photostream recommender system is proposed and evaluated. Flickr users tend to navigate through *photostreams*, *i.e.*, a set of photos that share some common characteristics such as same author, similar topic, shared group, *etc.* We first study how users exploit the photostreams. Then, different recommender systems are proposed in order to compare a standard tag-based approach with a collaborative filtering based on similar users navigation behavior. The content of this chapter is a result of collaboration with Luca Chiarandini, co-author of the following article:

- [31] Luca Chiarandini, Przemyslaw A. Grabowicz, Michele Trevisiol, and Alejandro Jaimes, "Leveraging Browsing Patterns for Topic Discovery and Photostream Recommendation", in *International AAAI Conference on Weblogs and Social Media (ICWSM2013)*, Boston, USA, July, 2013.

Chapter 7: In a news portal like Yahoo News with a vast collection of news articles with thousands of new articles posted every day, it is very easy for the user to get lost in the amount of data without reaching all the highly interesting articles. We consider the cases where the user profile is not known, in other words where we do not know the user's tastes. Our approach is to use the *ReferrerGraphs* to improve the articles to recommend to the user. The content of this chapter is based on the following article:

- Michele Trevisiol, Luca Maria Aiello, Rossano Schifanella and Alejandro Jaimes, "Cold-start News Recommendation with Domain-dependent BrowseGraph", *under review.*

Part III. Implicit and Explicit Information

In the last part we compare different explicit and implicit approaches. Among those, the ones based on the *ReferrerGraphs*, highlights the characteristics of each of them with the advantages and disadvantages.

Chapter 8: A study on user explicit action behavior is performed on Flickr, where we used users' favorite photos as a direct way of indicating a preference. The goal was to get insights about the "liking" behavior in social media, and to inform strategies for recommending items that users may like. Finally, we perform a favorite recommender experiment based on the outcome of our analysis. The content of this chapter is based on the following article:

- [69] Marek Lipczak, Michele Trevisiol, and Alejandro Jaimes, "Analyzing Favorite Behavior in Flickr," in *International Conference on Multimedia Modeling (MMM)*, Huangshan, China, January, 2013.

Chapter 9: Finally, we compare different approaches based on explicit actions, *i.e.*, favorites on Flickr, and on the vast implicit actions such as clicks, views and *ReferrerGraphs*. Results show how the rankings computed by different methods vary. Finally, thanks to the various evaluation metrics we used, the peculiarity and specific characteristics of each approach are highlighted and discussed. The content of this chapter is based on the following article:

- [100] Michele Trevisiol, Luca Maria Aiello, and Alejandro Jaimes, "Image Ranking Based on User Browsing Behavior," in *Special Interest Group of Information Retrieval (SIGIR2012)*, Portland, USA, August, 2012.

Conclusions and Appendix

The reminder of the thesis includes the conclusions and the appendix. The first one discusses the overall results and contributions presented in the dissertation, whereas the latter one contains the supplementary material and other information that might assist the reader.

Chapter 2

State of the Art

This section contains an overview of the most recent work related to this thesis. It covers work related to implicit user feedback, with a particular focus on user browsing behavior, *BrowseGraph*, and the cold-start problem in recommender systems. Related publications on recommendation and ranking, based on both implicit and explicit information, are also discussed.

2.1. User Browsing Patterns and BrowseGraph

In recent years, a large number of studies of user browsing traces have been conducted. Characterizing the browsing behavior of users, such as type of pages they want to visit and time spent, is a valuable source of information for a different number of tasks, ranging from understanding how search behavior difffers among people/users [116], ranking web pages through search trails [3, 117] or recommending content items using historical browsing data [101]. In addition, also user demographic information, such as age and gender, are deducible from the browsing traces [52].

In this thesis we focus on a data structure that is used to model selective browsing patterns, presented by Liu *et al.* [72, 73]. They introduced the *BrowseGraph*, a graph built by the users navigation patterns where the nodes are web pages and the edges are browsing transitions made by users. Liu *et al.* [74] discussed the difference between the standard hyperlinks graph of web pages, and the graph of the browsing data. They compared different centrality metrics computed on the standard hyperlink graph model, on the *BrowseGraph*, and on their combination, finding that the *BrowseGraph* top ranked pages have higher quality. Moreover, they show that a browsing

graph generated from about 15 days of data is stable enough to be reliable, in terms of convergence of centrality-based algorithms. As a result of its good performance, several variations and improvements of *BrowseRank* have been proposed in the recent past [127, 43, 30]. Besides the ranking algorithm used, the quality of the ranking is heavily influenced by the graph that is used to model the relations between documents. We want to remind that a centrality algorithm is a method that measures the relative importance of the nodes within a graph. In the next section we discuss some of the applications that are based on these types of algorithms.

In the experiments discussed in the thesis, we used the *BrowseGraph* for ranking and recommendation. We also proposed a novel graph, called *ReferrerGraph*, based on the *BrowseGraph* but characterized by sessions that start from the same referrer URL or domain. We found that the *ReferrerGraphs* are extremely valuable to identify the interest of the users through the referrer domain, and to recommend personalized items also in a cold-start situation. We are the first to experiment this type of graph for ranking and recommendation of media items.

2.1.1. Graph-Based Ranking Algorithms

Among the most popular algorithms to rank nodes in a graph [15], there are PageRank [81], HITS (Hypertext Induced Topic Selection) [58] and SALSA (Stochastic Approach for Link-Structure Analysis) [61]. In addition, there are many extensions of PageRank, such as TrustRank [48], VisualRank [56], and *BrowseRank*. In this latter one, described by Liu *et al.* [72, 73] and computed on the *BrowseGraph*, the web pages are weighted not only by the number of incoming and outgoing links, but also by the time that users spend on each page. We used this algorithm in Flickr, based on the assumption that more time the users spend watching an image and stronger is the implicit interest they express. In our experiments, we modify the original *BrowseRank* algorithm, by computing (instead of estimating) the exact values of *stop* and *reset* probabilities (see Section 9.2.2 in Chapter 9).

Traditionally web ranking algorithms have also been used to rank images. The most frequent application use the meta-data associated with the images, social network data of the authors of the images, and content-based approaches. A number of alternatives has been explored to improve the result of the web ranking task, including visual diversification [106, 118], near duplicate detection [35], query expansion [51], visual position [33], faceted detection [108], and re-rank based on click data [54]. Many works have ap-

plied PageRank to multimedia retrieval. For example, a solution presented by Jing *et al.* [55] is based on the content of the images: they extract visual features from each image, classify the common ones, and create visual links between them. In that case, the hyperlinks are given by visual similarity among the visual features of the images. Liu *et al.* [70] instead, use a random walk algorithm for tag ranking in Flickr, to overcome the sparsity problem of the tags associated to the images inside the social network.

In the experiments described in this thesis, we use the *BrowseGraph*, and we are not aware about any work that exploit this graph to understand the user behavior, in terms of which page the users visit and which content they prefer to consume. We extend the concept of *BrowseGraphs*, building, analyzing, and exploiting the *ReferrerGraphs* for the purpose of recommendation. Surprisingly, the informativeness of the referrer URL (or domain) in a user browsing session has been studied very little. Figueiredo *et al.* [42] and Yang and Leskovec [122] analyze popularity of content in online media. They show that the referrer has a strong influence on the popularity of items and could be used to predict it. Although not related to user browsing, they acknowledge the importance of the referrer domain. A closer work was presented by Ratkiewicz *et al.* [87], where the traffic of Wikipedia articles was studied, in terms of entry points, discovering that 95% of incoming traffic was done by other Wikipedia pages or search engines. However, they did not analyze any further this resource (*i.e.*, referrer URL) and they did not experiment any applications.

We are not aware about other studies that investigate the relation between the referrer URL and the characterization of the user browsing session. In this thesis we analyze in depth the referrer URL, building *ReferrerGraphs* and proving how these new graphs might help to identify the users interests, also in a context of cold-start.

2.1.2. Local Ranking Problem

It is important to report, that PageRank-like algorithms applied to a complete network or to any of its subnetworks, yield very different results. This problem is defined as the *PageRank Local Ranking Problem* (LRP) [14, 18]. The LRP was first introduced by Chen *et al.* [29] in 2004, who addressed the problem to approximate/update the PageRank of individual nodes, without performing a large-scale computation on the entire graph. They proposed an approach that can tackle this problem, by including a small number of nodes in the local neighborhood of the original nodes. Furthermore, Davis

and Dhillon [37] estimated the global PageRank values of a local network using a method that scales linearly with the size of the local domain. Their goal was to rank web pages in order to optimize their crawling order, something similar to what was done by Cho *et al.* [34] who instead selected the top-ranked pages first. In contrast with Boldi *et al.* [13], they found that crawling pages with highest global PageRank perform worse in terms of convergence time (to the global rank values). In this work we partial expand the local graph with the neighbors nodes with highest (local) PageRank showing an initial improvement on the convergence speed.

In 2008 the problem was reconsidered by Bar-Yossef and Mashiach [8] where they simplified the problem calculating a local *Reverse PageRank* proving that it is more feasible and computationally cheaper, as the reverse natural graphs tend to have low in-degree maintaining a fast PageRank convergence. Bressan and Pretto [18] proved that a general, efficient local ranking algorithm does not exist, and in order to compute a *correct* ranking it is necessary to visit at least a number of nodes linear in the size of the input graph. They also raised some of the research questions that we discuss in Section 5.4.2. They reinforce their findings in later work [17] where they summarized two key factors necessary for efficient local PageRank computations: *exploring the graph non-locally* and *accepting a small probability error*, two constraints that we are also considering in order to perform our experiments on the browsing graphs.

When one wants to estimate PageRank in a local graph, the problem of the information missing is tackled in various ways. In [8, 18] for example, the authors make use of a model called *link server* (also known as *remote connectivity server* [11]) that responds to any query about a given node with all the in-coming and out-going edges and relative nodes. This approach with the knowledge about the LRP allows estimating the PageRank ranking, or even the score, with the relative costs. A similar problem was studied by Andersen *et al.* [4], where their goal was to compute the PageRank contributions in a local graph motivated by the problem of detecting link-spam: given a page, its PageRank contributors are the pages that contribute most to its rank. Contributors are used for spam detection since you can quickly identify the set of pages that contribute significantly to the PageRank of a suspicious page.

In our case, the Local Ranking Problem might occur because the subgraphs that we use (*e.g.*, Flickr, Yahoo News) are subsets of the entire Web. Therefore, a ranking performed by PageRank-like algorithms applied to the entire

Web, might present very different results compared to the same algorithms applied to our subgraph (*i.e.*, Flickr browsing graph). We study, for the first time, the Local Ranking Problem on the *BrowseGraph*, with the aim to understand its reliability for our usage. The problem we consider here is different and largely unexplored because we are studying the PageRank of the different subgraphs based on user browsing patterns. None of the previous work have tackled the *Local Ranking Problem* as we do in this thesis (see Chapter 5), for various aspects: *a*) we study this problem for the first time on a browsing graph; *b*) we perform the "growing ball" experiment, namely expanding the subgraph incrementally adding nodes, and we study the convergence of the PageRank scores at each step; *c*) we compare different nodes selection approaches, and we found that adding few but very representative nodes, leads to obtain a yet satisfactory PageRank convergence, minimizing the final size of the graph. We use these findings in Chapter 9, where we expand the local (browsing) graph with the external referrer nodes, that are very representative (for centrality-based approaches) due to the high traffic made by the users.

2.2. Flickr: Browsing, Ranking and Recommendation

Flickr is a photo-sharing social network where users can share content and interact among them. It has been used extensively in research, in large part because it provides a public API that has allow researchers to easily obtain large datasets.

2.2.1. Browsing Photo Collections

Several studies investigate image browsing patterns of users. Lerman *et al.* [63] jointly use information about tags, social network, photo groups and photo views to understand how different people browse photos. Srikant and Yang [96] use implicit information extracted from server logs to improve the design of a website. In particular, the authors analyze the server logs in order to suggest modifications to the website link structure, to make content easier to find for the users.

Various interfaces have been considered for image browsing. Fan *et al.* [41] describe *JustClick*, which recommends images via interactive exploratory search. They build a topic network based on Flickr tags, and propose an interactive interface that allows the user to express a query by selecting im-

ages. Their experiments are performed on a big Flickr dataset of 1.5 billion images with 4,000 different topics. Xu *et al.* [121] present an innovative visual search interface based on topic clustering. Given the query and the results from a search engine, latent topics are detected and clustered, then, the clusters are shown in an intuitive layout. Ren and Calic [88] present an interactive interface for browsing of large-scale image collections. Their system is based on two main parts, an image clustering module and an interface generation component, in order to retrieve the images in a more efficiently way. Strong *et al.* [97] presented an approach for browsing images based on conceptual and visual similarity, with the main benefit being that the displayed images are grouped together. Zavesky *et al.* [126] proposed a new framework called *Visual Island*, a novel organization algorithm for interactively displaying results. The aim is to organize the images in order to improve human comprehensibility and reduce required inspection time.

Multiple studies investigate image similarity in photo collections, *e.g.* [21, 125], where the goal is to organize the images that present similar visual or textual information in groups of the same topic. We use user-generated photostreams instead and split them into batches of photos. In the case of Flickr, Gozali *et al.* [45] used a hidden Markov model to split photostreams into groups of similar photos, and evaluate different layouts to represent them.

In Chapter 6, we compare a tag-based recommender with a collaborative-filtering approach based on previous sessions of users. We evaluate them with a user study, considering characteristics such as serendipity and novelty.

2.2.2. Ranking in Flickr

Much work has been done for image ranking in the state of the art. Particularly, in Flickr, the majority of the cases focus on the images with the highest number of favorites. A "favorite" is a strong ground truth of positive user preference: it is the most explicit action that users can make to show their interest for a specific item. The favorite in Flickr, have been widely studied and used in research. Pedro *et al.* [84] used the number of favorites as relevance values for building and testing machine learning models. There are also studies that aim to detect favorite photos in Flickr or to predict the photos that a user is likely to favorite based on social, visual, and textual information [109]. Other papers investigate explicit and implicit features that lead to similar results of the favorites. For instance, Prieur *et al.* [85] find a very high correlation between the number of favorites and the number

of comments and views. Nevertheless, the reasons to select a picture as a favorite can be many[1], and they do not depend always on the user's interest. We show that using explicit features as favorites in an environment as Flickr, often leads to specific types of results since these actions are biased by the social network itself (see Section 9.3).

The quality evaluation of any ranking of multimedia objects is not a trivial, task due to the many quality dimensions at play. The attractiveness [44] and aesthetics [49, 57] of images retrieved, always play an important role in user's satisfaction. Lerman *et al.* [65] for example, propose an automatic method to assign photo attractiveness values to photos, by using textual and visual features. But in a specific environment such as an image-sharing social network, the relevance of the results may depend strongly on the social interactions. As many previous studies suggest [64, 22], social browsing and contact relationships are very important to model the interestingness of a resource in a socially connected environment.

In this thesis we exploit the *BrowseGraph* to compute an alternative ranking of images, based on the navigation patterns of the users. We used a modified version of the *BrowseRank* adapted to the Flickr browsing graph. Our *BrowseGraph*-based ranking returns images that are widely visited (due to the browsing patterns), interesting (due to the *BrowseRank* that keeps into account the time spent) and with a strong visibility also outside Flickr (due to the use of the referrer URLs). In short, although the literature in ranking is vast, we are not aware of work that specifically examines the ranking mechanisms we analyze in this thesis (see Chapter 9), and study those in relation to the image ranking task. We evaluate the ranking approaches on difference prospectives, such as the internal (within Flickr) and external impact of the images, the reachability of the photos through search engines, the PageRank scores assigned directly by Google, and so on. Our aim is to highlight the different characteristics of the algorithms that we compare.

2.2.3. Photo(streams) Recommendation

The research in the field of recommender systems has been extensively studied in the last years [1, 89, 50]. A main classification divides recommender systems into content-based and collaborative filtering [1, 89]. In our experiment, in Chapter 6, we use both approaches to recommend photostreams (set of images) in Flickr. However, in general, both methods present several

[1] http://www.flickr.com/photos/pagedooley/6246688704/

limitations. Content-based systems do not consider the opinion of the users, whereas collaborative methods require a critical mass of user traces to provide meaningful recommendations. Mobasher *et al.* [79] proposed a system based on aggregate usage profiles consisting of clustered user transactions. A natural difference with our experiment is that we consider photostreams as explicit content and structured units. Furthermore, Herlocker *et al.* [50]) list the recommendation of item sequences as one of the possible goals for a recommender system. Nevertheless, this has not attracted much attention of researchers [20]. In this thesis, we recommend sequences of photos belonging to recommended photostreams in a two-level recommender system. The state of the art has never considered the photostreams as atomic unit of content. In our work, we consider photostreams as items, and we exploit previous user navigational patterns (among photostreams) to recommend these atomic units to new user based on her current navigation.

2.3. Cold-Start Recommendation

Cold-start problem recommendation refers either to new items or to new users. In this thesis we focus on the latter case, using the browsing graph (*i.e.*, *BrowseGraph*) with a particular focus on the referrer URL (*i.e.*, *RefererGraphs*), to personalize the content shown to these users. In literature, some standard solutions require an initial, even though small, set of preferences about the newcomer [95, 91, 2]. This situation is also known as *warm-start* scenario. The user profile, can be populated by asking to the users to rate a set of items first, or importing their preferences from external systems [86]. When a minimal user profile is available, a common approach is to apply association rules in order to expand the user profile. Sobhanam *et al.* [95] used clustering algorithms to find the most similar known items and then they extended the profile of the users with item related to their tastes. Shaw *et al.* [91] used also non-redundant rule sets showing how they can improve the results. Yang *et al.* [123] presented a Bayesian-inference based recommendation system, that exploit the social network structure of the user in order to perform personalized recommendation.

News Recommendation

Despite the progress of recommender systems in general, news recommendation is still a very active area[2]. The majority of news recommender systems

[2] http://recsys.acm.org/recsys13/nrs/

are based on user information collected over time [71, 36, 67, 68], relying on logged-in users only and combining collaborative filtering and content-based approaches. However, when reading news website, the user is most often not logged-in, making impossible (or at least very difficult) the creation of a reliable user profile. There are approaches based on models of similarity among news articles [76, 67, 90], that consider different key factors such as textual similarity, recency, coherence, novelty and popularity. Tsagkias and Blanco [102] proposed a language intent model that extract a query from the user browsing session. In this way they model the user interest using the queries of the users. There are also approaches looking for the most authoritative news sources as a proxy for quality [39, 98]. To perform an efficient news recommender, the user taste has to be considered as they might change over time. Indeed, studying the users browsing patterns, Liu *et al.* [71] shown that more recent clicks have a considerably higher value to predict future actions that the historical browsing record. Through their experiments, they conclude that the best results are obtained by combining user's genuine news interests with the current news trend in her location, to generate personalized news recommendations.

In Chapter 7, we perform news recommendation facing the cold-start problem of newcomers. In our experiment, we consider only the news article the user is currently visiting, and the referrer URL of her session. We are not aware about any other work that exploits the referrer URL directly as main information for a recommender system. Most of previous research relies on the user historical profiles. Our work is different also because it is based on the *BrowseGraph*, and on the previous collected users' behavior, grouped by the external domain from where they come from (*i.e.*, *ReferrerGraphs*).

Chapter 3

Preliminaries

3.1. Methodology

This section explains the generic approaches used in the experiments described in this thesis. The main goal is to extract meaningful information from users' browsing patterns in order to gain insights about their preferences. We are considering a particular type of users known as newcomers, *i.e.* users completely unknown to the system, so there is no information about what they are looking for and what they would like to consume. However, once newcomers enter the website, they leave implicit information that we can analyze in order to understand their preferences.

- We collect all the users' navigation patterns, *i.e.* the sequences of web pages visited, and we build a weighted graph based on these sequences. In this graph the nodes represent the web pages, the edges represent the transitions and the weights are based on the number of times the transitions have been done. This graph is called a *BrowseGraph* [72].
- One information that has never been studied much in the literature is the external referrer URL, *i.e.* the last page visited by the user before visiting the current website. For example, when a user clicks on a link of a Flickr photo on Facebook, he gets redirected into the Flickr website. In this case, from the Flickr logs, it is possible to see the navigation pattern of the user knowing that he comes from Facebook. Simplifying, we group the external referrer URLs into their domains (*e.g.*, *www.facebook.com/xyz* into *Facebook*), and for each of the most frequent domains (*e.g.*, Facebook, Google, Twitter, Yahoo) we add a

node to the graph. In this way, the final *BrowseGraph* contains also the most common entry points, that we call external nodes because they do not belong to the Flickr network. These nodes could be decisive to infer the importance of Flickr pages that have been linked on external web pages. Moreover, users coming from the same external node (*e.g.*, Twitter), might be more interested to a certain type of content with respect to users coming from other entry points (*e.g.*, Facebook).

- In order to exploit directly these external nodes, we build different graphs based on the most frequent external referrer domains and we call them *ReferrerGraphs* (we are not aware about any previous work that built or used similar type of graphs). These graphs contain the behavior of the users coming from different domains, and they contribute to predict what type of content the users are going to consume.
- Once we have these graphs, we can apply centrality based algorithms that infer importance of the different web pages, and since we included in the graph also the external nodes, these will be taken into account in the computation. We slightly modified the well know PageRank and an extended version called BrowseRank [72]. Once we have the ranking of the nodes, we can promote the most interesting content (*i.e.*, web pages) to the users depending on which external domain, *i.e. ReferrerGraph*, they are coming from.

The methodology presented in this thesis has many advantages. First, the required datasets (*i.e.*, web logs) are available to each service provider. Second, the methodology can be adapted to any domain and context. Our experiments are made on a photo-sharing website and on a news portal. Finally, it shows a different solution to personalize interesting content to newcomers, dealing with one of the most challenging problems of personalization and recommendation.

In the next sections, we describe the datasets and the approaches used in the experiments of this thesis.

3.2. Data Types and Processing

This section introduces the data sources commonly used in this thesis. The main datasets are based on the user browsing logs of Flickr and Yahoo News. In addition, this section explains the process to extract the sessions and to construct the *BrowseGraph*, widely used in the rest of the thesis.

3.2.1. Browsing Log Data

For the purpose of this thesis we consider user-anonymized log data. The data is comprised of a large number of pageviews, which are represented as plain text files that contain a line for each HTTP request satisfied by the web server. For each pageview in the dataset, we gathered the following fields:

$$\langle BCookie, Time, ReferrerURL, CurrentURL, UserAgent \rangle ,$$

The *BCookie* is an anonymized identifier computed from the browser cookie. This information allowed us to re-construct the navigation session of the different users. For the purpose of this study, in general we ignore any other user related information such as IP address or demographic data. *CurrentURL* and *ReferrerURL* represent, respectively, the current page the user is visiting and the page the user visited before arriving at the destination page. Note that the *ReferrerURL* could belong to any domain, *e.g.*, it may be external to the Yahoo News website. The *User-Agent* identifies the browser in use, and *Timestamp* indicates when the page was visited. All the data is anonymized and aggregated prior to building the browsing graphs. We removed traffic derived from web crawlers using the information contained in the User-Agent field. Essentially, we preserve only the entries that contain a well-known browser identifier (*e.g.*, Explorer, Chrome, Firefox, Safari and Opera).

3.2.2. User Sessions and BrowseGraph

The structure of a website is typically represented as a graph where nodes are pages and edges are the hyperlinks connecting them. In this model all the links have the same weight, disregarding how many times users go through them. The BrowseGraph [72, 73] is an alternative representation that captures the importance of the user navigation patterns by considering the actual transitions from one page to another, rather than hyperlinks. The *BrowseGraph* is a graph whose nodes are web pages and whose edges are the browsing transitions made by users. We build a weighted graph, where the weight of an edge represents the number of times users made that transition. To build it, we extract the transitions of users from page to page, and in order to preserve the user behavior (that could vary over time), we group pageviews into *sessions*. A session consists of a sequence of pageviews made by the same user in a *continuous* segment of time. To break the sequence of page visits into sessions we account for the time between them, as

transitions far apart in time will likely not follow any logical browsing session relation. In other words, we need to identify when the user interrupts her navigation since when her navigation is restarted, the intentional behavior will be different. We split the activity of a single user, taking the *BCookie* as an identifier, into different sessions when either of these two conditions holds:

- **Timeout:** the inactivity between two pageviews is longer than 25 minutes. This is a standard value used in research [24] as well in production systems.[1]

- **External URL:** whether two pageviews are visited by the user from different referrer URLs, the current session ends even if previous visits are within the 25 minute threshold. The assumption is that, if the user enters the website from different external referrer URLs (*e.g.*, first from Facebook then from Twitter), the user's interest might be different.

3.3. Data Sources

As we mentioned, the experiments performed to validate this thesis used different datasets, mainly Flickr and Yahoo News. In this section the common data sources are described, whereas any other specific dataset is discussed later in the corresponding section. In addition, common pre- and post-processing steps on the data sources are explained, such as filtering, session building and URL categorization.

3.3.1. Flickr Browsing Data

The Flickr implicit log data is used in Chapters 4, 6 and 9. For the purpose of this study, we took a sample of the *pageviews* (see Section 3.2.1 for details) of more than 10 million anonymous users from approximately two months of Flickr user log data, from August to October 2011. All of the data processing was anonymous and aggregated. Flickr allows users to set specific pages to "private", hence our analysis considers only public pages.

[1] https://support.google.com/analytics/answer/2731565

Pageview Filtering and Data Selection

In order to obtain a coherent dataset in terms of both timezone and activity, we focused on users who were located in the USA by extracting the location of the IP address from the source of the HTTP request and filtering out non-USA locations. In spite of this filtering, and the ones described in Section 3.2.1, there are cases in which the User-Agent field indicates that a legitimate browser was used, but the corresponding "users" have a very large number of pageviews. The frequency, however, suggests that such server requests could not have been made by humans, but instead were done automatically for malicious crawling. We therefore apply an additional filter by which we set a maximum threshold on the total number of pageviews per user. The threshold is set to remove a small percentage of the users (1% of the total amount). After applying the filtering steps described above, our sample contains approximately 309 million pageviews.

Source URL Taxonomy

In order to analyze the referrer URLs (*i.e.*, the websites users arrive to Flickr pages from), we built a taxonomy for external URLs (*i.e.*, whose domains are different from `www.flickr.com`). The first attempt of categorizing URLs was based on the Open Directory Project[2] and the Yahoo Directory.[3] However, by manually inspecting the results, we realized that the classification was too detailed and did not capture the aspects we are interested in. In fact, URL categorization usually works by *topic* (*e.g.*, travel, economy, food, *etc.*) whereas in our study we are interested on a categorization by *type* (*e.g.*, blog, social networking site, search, *etc.*).

We therefore opted to annotate them manually (*e.g.*, `search.google.com` as *search*, *etc.*) and focused on defining 17 categories that we considered important. We created a set of regular expressions in order to identify about 210 different external URL domains. Table 3.1 shows the most frequent source categories, clearly showing that the distribution is quite skewed.

Building the Flickr BrowseGraph

The basic idea in our approach is that the navigation patterns within a social media platform have a strong impact on the importance of content. Therefore, we build a *BrowseGraph* as explained in Section 3.2.2, based on

[2]Netscape (AOL), "Open directory", `http://www.dmoz.org/`, June 1998.
[3]Yahoo, "Yahoo! directory", `http://dir.yahoo.com/`, March 1995.

Category	Examples of Content	%
search	search.yahoo.com, google.com, *etc.*	34.87
social	facebook.com, tumblr.com, *etc.*	26.95
mail	mail.yahoo.com, gmail.com, *etc.*	13.22
aggregator	reddit.com, stumbleupon.com, *etc.*	7.76
blog	blogspot.com, blogger.com, *etc.*	6.65
photo	flickrhivemind.net, compfight.com, *etc.*	2.32
microblog	twitter.com, *etc.*	2.26
forum	discussions, forums	2.00
news	news.yahoo.com, cnn.com, *etc.*	1.67
shop	ebay.com, *etc.*	0.85

Table 3.1: Top ten most frequent source categories in the dataset. For the entire list see Table A.1 in Appendix A.1.

our Flickr data but considering as nodes only certain types of web pages. We create one node of the graph for each pageview that refers to one of three *entities* in Flickr, namely *users*, *groups*, and *photos*. Figure 3.1 shows examples of pages that are mapped to each entity, noting that all pageviews that show the same entity are condensed into a single node in the BrowseGraph. In other words, a single node represents several types of pages. For instance, since various URLs show the same image, all of those are mapped to a single node (fullscreen, slideshow, *etc.*).

The motivation behind this representation is that we are interested in ranking *the photographs* and those photographs may appear in multiple places (*e.g.*, the photo appears prominently in the photo page, in the slide show, and in the photo favorites layouts of Figure 3.1), but since we are not interested in ranking each of the individual pages, we group them into a single node. Flickr contains other page categories (*e.g.*, personal settings, photo upload pages, mailbox) which we do not consider, so we refer to them as non-entities and we do not create any nodes for them. The main reason to do so is that we are interested in the navigation between entities in Flickr. Therefore we need to discard some categories of traffic: *navigational* (*e.g.*, searching for photos), *configurational* (*e.g.*, changing settings, profile information) or *messaging*.

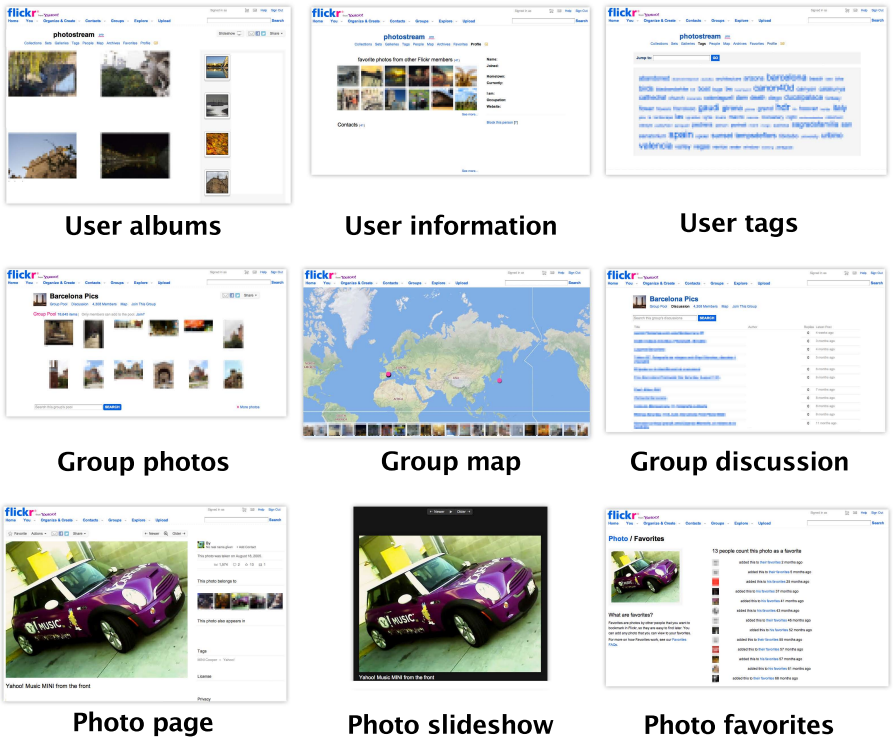


Figure 3.1: An example of pageviews that correspond to the entities of the Flickr *BrowseGraph*. Each row corresponds to an entity from top to bottom: user, group and photo.

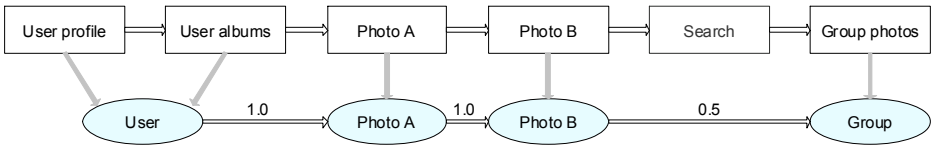


Figure 3.2: An example session is illustrated at the top, and the corresponding derived *BrowseGraph* is shown at the bottom. Gray arrows display the mapping between pageviews and *BrowseGraph* nodes.

To build the *BrowseGraph*, we create the subgraph of each session and we then merge all subgraphs. Given a session $s = (p_1, p_2, ..., p_N)$ where p_n is a pageview, we map each entity pageview p_n to the vertex of the *BrowseGraph* and we connect them in the order that they appear in the session. We then

weight the arcs according to the number of non-entity pageviews between the source and the target. Intuitively, we would like to give the highest weight (namely 1) to the arcs that connect entities that appear in consecutive pageviews and a lower weight to pageviews that are more distant, to better express their actual proximity in the browsing activity. For example, in Figure 3.2, "Photo A" and "Photo B" are closer in the session than "Photo B" and "Group". We do so by assigning the weight $w_{ij} = \frac{1}{NE(i,j)+1}$ where $NE(i,j)$ is the number of non-entity pageviews between i and j. We then compute the *BrowseGraph* by summing up all arcs with the same source and target.

A fragment of *BrowseGraph* is shown in Figure 3.2. The top row shows pageviews in a session, and the bottom part shows the resulting *BrowseGraph*. A vertex is created for each entity present in the session and gray arrows represent the mapping between pageviews and vertices. We can observe that the *Search* pageview, that displays the results of a query of the user and therefore does not refer to an entity, is not mapped to any *BrowseGraph* vertex but influences the weight of the corresponding edge of the BrowseGraph.

Modeling accesses from the Web (*i.e.*, domains different from Flickr) is important to detect the most frequently accessed entities from external sources. We therefore include as nodes in the graph, also the external referrer URLs, belonging to the sessions that we consider, from where the users enter in Flickr. However, since we are interested in understanding global navigation patterns, the full URL is too specific. For this reason we group external URLs in 17 categories as it is described in Section 3.3.1, and some of them are shown in Table 3.1. These categories cover around 99% of the total number of external URLs. For each category, we add a node to the *BrowseGraph* and we connect it to the nodes that correspond to the first entity of a session coming from the category.

3.3.2. Yahoo News Browsing Data

The Yahoo News Log Data is used in Chapters 5 and 7. For the purpose of this study, we took a sample of Yahoo News network's[4] user-anonymized log data collected in 2013. The data is comprised by a large number of pageviews, that after applying the filtering steps described in Section 3.2.1, contains approximately $3.8M$ unique pageviews and $1.88B$ user transitions.

[4]We considered a number of different sub-domains like *Yahoo News*, *Finance*, *Sports*, *Movies*, *Travel*, *Celebrity*, *etc.*

Session Identification and Characteristics

In our dataset we are able to associate to each pageview additional information, like *time spent* by each user visiting that page, and a *category* label that uniquely identifies which category the page belongs to. The latter one, allows us to understand the type of content of the web page; some results are displayed in Table 7.3, that shows which are the most visited categories broken down per referrer URL.

ReferrerGraph: Sub-Graphs Based on Session Referrer domain

We aim to compare the behavior of different users that access the Yahoo News network from different referrer domains. We first consider users accessing the *news.yahoo.com* portal directly from the homepage. We further consider a number of domains that fall outside of the Yahoo News network. In particular we tracked the following referrer domains: *search engines* (Bing, Google, Yahoo), and *social networks* (Facebook, Reddit, Twitter). For each referrer domain we extract a subgraph of the overall *BrowseGraph* by considering only the browsing sessions whose initial *r*eferrer URL matches that domain. For example, if a user clicks on a link referring to our network that has been posted on Twitter, her *r*eferrer URL will be the Twitter page where she found the link. Next, we consider all the following pageviews belonging to the same session of the user as being a part of the *twitter-subgraph*, given that all of them have been reached through Twitter.

We applied the same procedure for all the referrer domains that we defined above, and finally we obtained a weighted graph for each different external URL with the following structure:

$$\langle Pageview_{source}, Pageview_{destination}, Weight \rangle \ ,$$

where *Weight* accounts for the number of times a user has navigated from the source page to the destination page.

Part I

Browsing Activity and BrowseGraph

Chapter 4

User Browsing Activity

The analysis of user browsing behavior gives us indications of the taste of users and their engagement within the website. The way users navigate might tell us, for example, how fast they are browsing and spending time in each page, or whether they are sharing the content through some social platform. Knowing which pages the users have visited means being aware of the topics and the content that capture their interests, and estimate their implicit preferences. However, one information that has not grabbed much attention in the research community is the *external referrer URL*, namely the URL visited by the user before entering in the current website. The external referrer URL contains some initial information that might be very helpful in order to characterize the user's interests, especially for the newcomers in the cold-start case. In this chapter, we analyze the browsing patterns of users with respect to the external referrer URL, used as *referrer URL* or just as *referrer*. The results of this chapter were published in [32].

4.1. Introduction

Insights into how users behave within a website or domain are extremely important in informing business decisions, in developing strategies to provide new functionalities, and in general for devising new algorithms that directly improve such services. For instance, information on the most visited pages or sections can be used not just to create better user models, but also to improve the design of such pages and the overall "flow" of the website (*e.g.*, by highlighting certain sections on particular page layouts), or simply on particular layouts).

Flickr has become a rich resource for research in multimedia, mainly due to its clear copyright policies and APIs, which have facilitated the gathering and analysis of Flickr data. A lot is known about the data that resides in Flickr. But how do people actually use Flickr? And in particular, which social navigation patterns do they follow?

As the functionality of the Web has become more complex, the sharing of the content such as Flickr photos is done in multiple ways, for example, by posting to social networks such as Facebook, to information networks such as Twitter, or to (personal) blogs. As a consequence, it has become increasingly more difficult to understand the dynamics of how users browse and look at photos once they arrive at Flickr from other sources.

This chapter aims to address several questions, including but not limited to: *(i)* whether photo social navigation patterns differ depending on the referrer, and if there are differences, what kinds of differences there are, *(ii)* whether similar types of websites, such as "search", lead to similar behavior, *(iii)* what are the types of pages, within Flickr, that are more popular depending on the referrer, and *(iv)* whether user behavior, in terms of time spent, varies depending on the referrer.

The results of the experiments in this chapter, show the ability to characterize the type of session that the user is doing based on the referrer domain. These outcomes will be used later as the basis for developing applications such as recommendation and ranking.

4.2. Dataset and Session Analysis

We analyze the Flickr user logs applying the pageview filtering described in Chapter 3 (Section 3.3.1), obtaining approximately 309 million pageviews. Moreover, since the purpose of this study is to understand if there is a

relation between the type of page the user is interested to visit and the referrer URL, we define a categorization of the Flickr web pages called *page layouts*, or simply *layouts*. Accordingly, we also classify the referrer URLs using the taxonomy described in Section 3.3.1.

4.2.1. Pageview Layouts

In most websites, multiple URLs can map to exactly the same page "layout". For example, on Flickr, the URL of a page that shows a single image contains a unique ID for the image,[1] thus two URLs for two different images are different[2] even though the page layout is the same. Since our interest is in modeling navigation patterns in Flickr, we must map all URLs that refer to the same layout, to a single page layout (*e.g.*, "single image page"[3]). For this purpose, we define the *page layout*: a hierarchical taxonomy of URLs. We manually created a set of regular expressions to classify the URLs to obtain a total of 96 different page layouts. Example of layouts include the following: *display all user photos*, *search photos*, *browse group photos*, *add contacts*, *accept invitation to join Flickr*, *etc.* The entire list of page layouts see Table A.2 in Appendix A.2.

4.2.2. Session Characteristics

The sessions are identified and processed as described in Section 3.2.2. Table 4.1 shows some statistics computed over aggregate sessions in our sample dataset. The last two rows show the number of different types of page layouts present in the sessions. The values suggest that a large number of sessions tend to consist of only a few page layouts. It is important to note, however, that a more "complex" use of Flickr is not uncommon, and represented by sessions in which a maximum of 39 different page types are visited.

4.2.3. Analysis of Types of Pages Visited

Table 4.2 shows the ten most visited *page layouts* in the dataset. We can see that there are a few page layouts that are visited most frequently: although we defined a total of 96 page layouts, users tend to navigate through a small subset of them, namely to explore *photos of users* and *groups*. This is compatible with the results of Table 4.1 that shows that users usually browse in just a few layouts during one session.

[1] PhotoId 14043395432: https://www.flickr.com/photos/xarabas/14043395432/

[2] PhotoId 7756641062: https://www.flickr.com/photos/katia_romanova/7756641062/

[3] URL for a page layout: https://www.flickr.com/photos/<username>/<photoId>/

Total number of sessions	40,446,676
Total number of users	10,912,431
Avg. number of distinct page layouts	1.83
Max. number of distinct page layouts	39

Table 4.1: Basic statistics about the sessions in the dataset.

Page Layout	**Description**	%
Display all user photos	Displays the photos of a user on a grid	26.71
Browse user photos	Displays full-page photo of a user and allows browsing to the next and previous photos	20.67
Browse user album	Displays full-page photo of an album of a user and allows browsing to the next and previous photos	14.12
Display single photo	Displays full-size photo	7.22
Homepage	Home page of Flickr	5.60
View user albums	Lists the album of a user	4.59
Browse group photos	Displays full-page photo of a group and allows browsing to the next and previous photos	2.63
Search photos	Photo search in Flickr	2.38
Browse user fav.	Displays full-page photo of the favorite photos of a user and allows browsing to the next and previous photos	2.09
Group photos	Displays the photos of a group on a grid	1.79

Table 4.2: Top ten most frequent page layouts in the dataset. For the entire list of page layouts see Table A.2 in Appendix A.2.

4.2.4. Referrer Categories Analysis

One of our main assumptions is that there is a relationship between the referrer URL and the type of navigation behavior of the user.

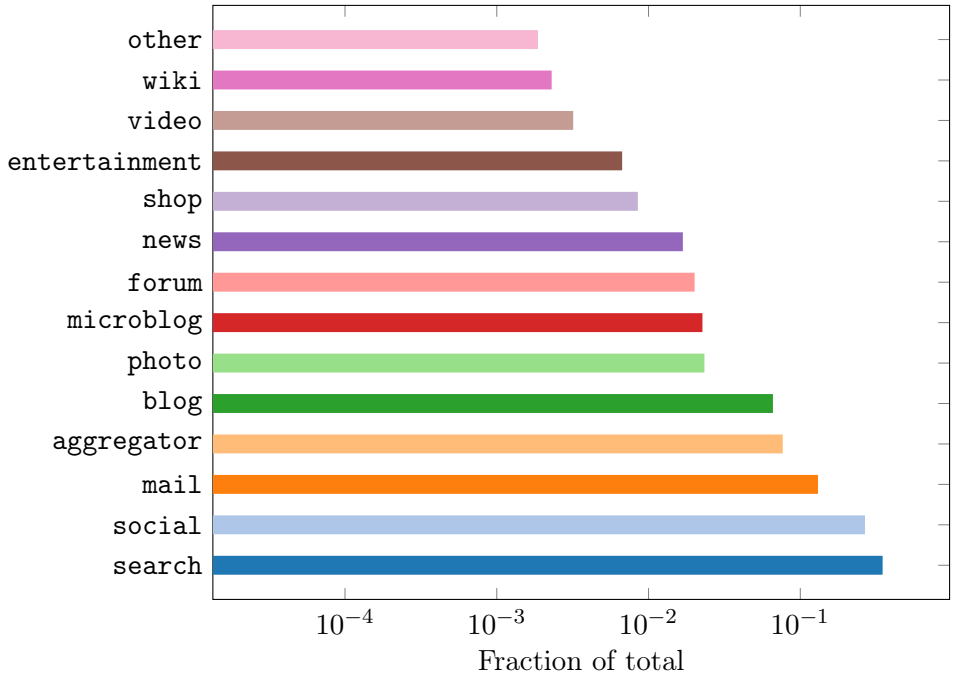


Figure 4.1: Distribution of the 14 categories for the external referrer URLs.

Recall table 3.1 in the previous chapter, where we showed the most frequent referrer categories from which the user arrives to Flickr pages. The histogram in Figure 4.1 shows the distribution of these categories. The two most common referrer categories are *search* and *social*. The presence of search is reasonable due to the contribution of image search and navigational queries. While most photo websites retain proprietary rights on the retrieved results, or do not have clear photo licensing policies, we can assume that Flickr is one of the main referrers of Creative Commons-licensed material[4]. Social network websites, such as Facebook, constitute very popular access points to Flickr since users are highly interested in photos shared by friends. We did not expected *mail* to have high importance, as usually the attachments are sent within the message itself and not as external links. As we will see in Section 4.3, many sessions derive from invitations of friends to join Flickr. The fact that many sessions come from the *news* domain is

[4]Flickr is well-known for the big amount of Creative Commons photos: http://www.blueglass.co.uk/blog/30-free-image-websites-creative-commons-royalty-free/

indicative that the image is often considered as appealing, or significant, as the actual text of the article.

This raw analysis gives us the first insights into how the initial context may affect navigation patterns. However, this is confirmed by observing the cumulative distribution of session lengths depicted in Figure 4.2. In the figure, we represent only the 9 most frequent categories, the whole list is shown in the Appendix in Table A.1. The categories have a different behavior from one another. The lines that reach value 1 sooner correspond to the situation in which the user spends less time on Flickr on average. On the contrary, the ones that grow slowly show users with longer sessions on average. Based on this analysis, we see that the shortest sessions originate from aggregators. One example is `www.reddit.com`, in which the links to Flickr appear inside news posts. It may appear strange that the sessions deriving from *news* are among the most lasting. An explanation for this might be that the visual material in news sites (such as Yahoo News), is curated by professional editors and photographers, and often consists not only of a single photo, but rather of a collection of photos related to a particular event. For example, an article about the earthquake in Japan is linked to a group or a set of photos all related to that topic. The user is therefore prone to see more than one picture.

An extreme behavior is observed in the *mail* category, where the users spend the longest time interacting with Flickr. One possible explanation might be that only the "closest" contacts send e-mail, and thus a stronger bond exists between the sender and the receiver of the message. Moreover, one could assume that users that share links via e-mail, may share entire sets or albums which contain many photos, leading to longer and more complex interactions with Flickr.

4.3. Clustering of Sessions

In this section, we describe the clustering of user browsing sessions. We analyze the general characteristics of the obtained clusters in terms of the page layouts that constitute the clusters, and in terms of browsing behavior depending on the referrer domain categories.

We model each session s as a vector $v = (v_1, v_2, ..., v_P)$ where each v_i counts the number of views of page layout i in session s. In order to compare the vectors we use the *cosine similarity* metric, since it is not affected by the

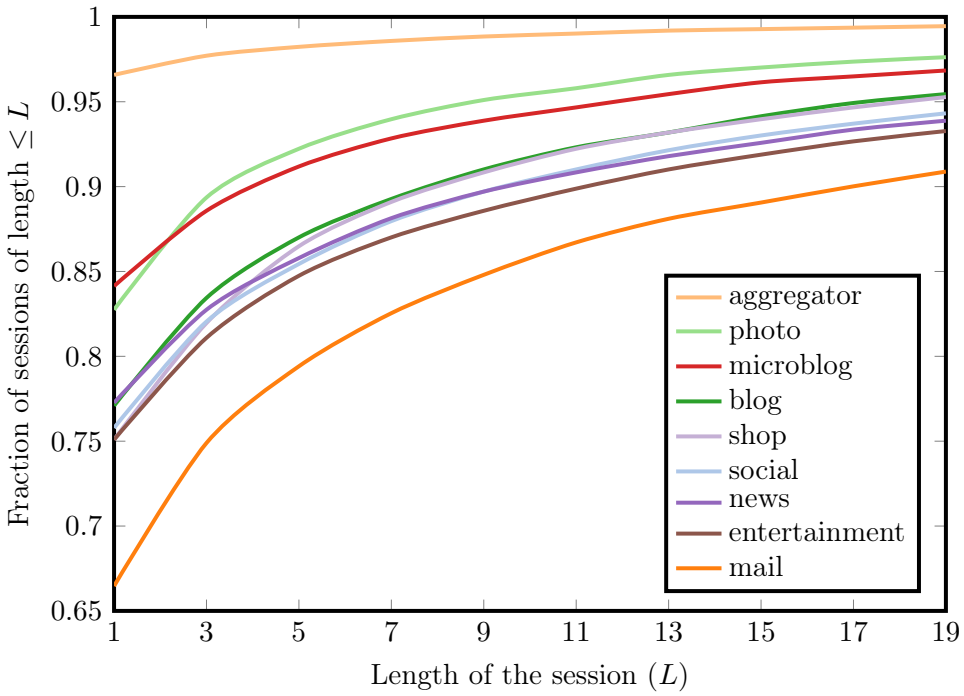


Figure 4.2: Cumulative distribution of the 9 most popular categories of referrer URLs.

absolute number of pageviews but only by the relative distribution across the page layouts. We apply the Canopy algorithm on the vectors, choosing empirically the parameters ($T1 = 60$ and $T2 = 40$) as a starting point to initialize the centroids. McCallum *et al.* [77] have shown how the Canopy algorithm decreases the computational time while also slightly increases the accuracy. Finally, we run the K-Means clustering to extract clusters of sessions, and we obtain a total of 62 clusters.

4.3.1. Patterns in Session Clusters

Although our hypothesis is that user browsing patterns might differ depending on the referrer, we first examine session clusters without taking into account how users arrived at Flickr. We will then remove this constraint in Section 4.3.2.

	c0	c1	c11	c24	c37	c51	c59
Browse user fav.	0	0.02	0	0	0	0	0
Photos of group	0	0	0	0.06	0	0	0
Browse user album	0	0	0.92	0	0	0	0.28
Browse user photos	0.05	0	0	0	0.06	0.38	0
View user albums	0.02	0.02	0.06	0	0	0	0.61
Search photos	0	0	0	0	0	0.02	0
User profile	0.02	0.03	0	0.08	0	0	0
Add contact	0	0.57	0	0	0	0	0
Display single photo	0	0.05	0	0.17	0	0.02	0
Group page	0	0	0	0.53	0	0	0
Display all user photos	0.43	0.08	0.02	0.05	0.81	0.46	0.03
Homepage	0.11	0.03	0	0.03	0.02	0.03	0.02

Figure 4.3: Heat-map of $p(layout|c)$ for the most frequent clusters. Darker squares indicate a higher presence of the relative pageLayout views (row) in the current cluster (column).

We want to extract general behavior of users browsing Flickr independently from the referrer URL. For this purpose, we focus on clusters that are generated in the same proportion by all the referrer categories. They capture actions that people do in Flickr that can be accounted as common use.

More specifically, we compute the entropy distribution for each cluster c across $p(c|\text{ref}_{\text{cat}})$, with the following equation ref_{cat}:

$$\sum_{\text{ref}_{\text{cat}}} [p(c|\text{ref}_{\text{cat}}) \log_2 p(c|\text{ref}_{\text{cat}})]$$

We then sort the clusters in ascending order and select some of the clusters with lowest entropy, finally picking 7 of them.. These clusters represent the common behaviors of users, in terms of page layouts, accessing Flickr in the same percentage for each referrer category. In order to understand the characteristics, we draw the heat-map of $p(layout \mid c)$ and the page layouts that constitute them, in Figure 4.3.

As the figure shows, c0, c37 and c51 contain a large number of *Display all user photos* and *Browse user photos* page views, which indicate a typical pattern of mainly browsing through the photos of a user (or users). Cluster c1, on the other hand, contains more cases of users that import and add new contacts (*Add contact* row in Figure 4.3). A very clear case of browsing photo albums is cluster c42, where we can observe a large value in the *Browse user album* row. A similar behavior is in cluster c59 where the sessions are more balanced between browsing a specific album (*Browse user album*) and seeing the list of albums (*View user albums*), maybe to explore a different one. Group-oriented navigation is specific of c24, due to the presence of *Group page* and *Photos of group*. In this case users switch between the main page of the group and its photos.

Although these clusters are useful to understand how users interact with Flickr, we would like to explore the peculiarities of the referrer categories. We therefore manually inspected the clusters and selected some of the ones that show interesting patterns.

4.3.2. Browsing from Different Referrer Categories

As stated earlier, many clusters illustrate a very specific browsing behavior. We manually picked a few of them to show how well they describe some navigation patterns in relation with the referrer categories (ref_{cat}). Figure 4.4a shows the distribution of such clusters across referrer categories, whereas Figure 4.4b shows the distribution of the same clusters across page layouts. Due to the large amount of sessions originated from search engines, the *search* referrer category appears in most of the clusters. Despite this, there are still some clusters in which this is not the case.

Cluster c24 shows a large contribution of *search* and *news*, and the distribution of page layouts for that cluster (first column in Figure 4.4b) is biased towards browsing of groups (*Group page*). This suggests that news editors embed sets of images into the article page. Moreover, photos of the same event are likely to be organized in the same group in Flickr. Cluster c58, slightly more evenly spread across all referrer categories, is similar to c24 but favors browsing through the photos of a group (*Photos of group*) on the home page of the group (*Group page*). Cluster c42 contains sessions coming from both *search* and *aggregators*, in which users visualize the tag cloud of photo tags used by another user. This tag cloud visualization, gives an aggregated vision of the content posted. Cluster c29, mainly originated from *search*, explores the list of favorite photos of a user (*Browse user fav.*).

	Cluster						
Referrer Category	c24	c58	c42	c29	c9	c33	c25
aggregator	0.01	0.05	0.25	0.06	0	0.02	0.82
blog	0.06	0.07	0.03	0.02	0.02	0.01	0.01
mail	0.04	0.03	0.02	0.06	0.01	0.84	0
news	0.23	0.04	0	0	0	0	0
photo	0.01	0.03	0.01	0.01	0.01	0.02	0.01
search	0.57	0.69	0.62	0.77	0.94	0.06	0.13
shop	0.02	0.01	0	0	0	0	0
social	0.04	0.05	0.05	0.07	0.01	0.04	0.01

(a) Heat-map of referrer categories $p(\text{ref}_{\text{cat}} \mid c)$.

	Cluster						
Page Layout	c24	c58	c42	c29	c9	c33	c25
Manage friends	0	0	0	0	0	0.29	0
Browse user fav.	0	0	0	0.41	0	0	0
Photos of group	0.07	0.5	0	0.01	0	0	0
Photo fans	0	0	0	0.02	0	0	0
Search CC photos	0	0	0	0	0.38	0	0
Browse user tags	0	0	0.4	0	0	0	0
Search photos	0	0.01	0.01	0	0.23	0	0
Add friend	0	0	0	0	0	0.33	0
Group page	0.53	0.07	0	0.01	0	0	0
Recent activity	0	0	0	0	0	0	0.7

(b) Heat-map of page layouts $p(\textit{layout} \mid c)$.

Figure 4.4: Heat-map of the most interesting clusters. Darker squares indicate higher values for the presence of sessions with that category (row) in the relative cluster (column).

Cluster c9 is another example of a cluster that contains sessions originated from search engines (last column in Figure 4.4a), and it is not surprising that Figure 4.4b shows that those sessions include mainly *search* pages inside Flickr. One assumption is that, in this case, users are migrating the search task to Flickr, in order to take advantage of the image search features, as for instance filtering photos by Creative Commons license or tags. Cluster c33 shows *mail* as a principal referrer category, and it is composed of page layouts related to social actions: *a*) *Manage friends* is the set of all pages related to adding, editing or removing information about contacts in Flickr; *b*) *Add friend* is the page in which the user is asked for confirmation when adding a contact. Manual inspection of the sessions suggests that the traffic in this cluster mainly derives from accepted invitation mails sent to mail contacts.[5] Sessions in cluster c25 are mainly originated from *aggregators*, and are aimed at checking the recent activity on the Flickr website (*i.e.*, recently added photos, albums, *etc.*). Indeed, the user to get an overview on recent events in external websites uses aggregators, including Flickr.

The remaining clusters have been inspected, but are not listed since they do not show interesting characteristics.

4.4. Summary and Discussion

In this chapter, we analyzed a sample from two months of Flickr user data, specifically on user logs. We classified pages within Flickr into specific categories, and analyzed how the behavior of users in viewing such page categories changes depending on the referrer domain (*i.e.*, the page they come from). Our analysis shows that there are important differences among users' social photo navigation patterns, and that the referrer domain largely affects these.

Our analysis showed that users arrive into Flickr from a variety of referrer domains, such as search, social, mail, and aggregators. We showed that the overall length of the sessions varies depending on the type of source domain. For example, users that arrive to Flickr from mail domains tend to spend more time than those arriving from any other sources. The distribution of visits from different types of sources, gives us interesting insights on the web as it is today (*e.g.*, social sites have a prominent place). Moreover, it was possible to make some observations on the behavior in terms of session

[5] We do not examine mail contents, so this hypothesis cannot be verified, and is based solely on the aggregate views of the "add friend" page.

length, for example, we noticed that users that click on mail links may be receiving photos from close social contacts, which might explain longer sessions. At the same time, we found that session patterns could be easily clustered. We found, for instance, that some sessions are very focused on viewing photos of users, while others focus on viewing photos in groups. Some of the common behavior can be intuitively explained, for example, sessions that originate in mail domains have a stronger focus on managing and adding friends.

Many similar observations can be made based on the figures presented in this chapter. Sessions that originate from search sites, for instance, cluster around the Flickr search functionality, suggesting that the user's main intent is indeed finding images of some sort. It's important to keep in mind, however, that such observations constitute hypotheses that need to be further examined.

In the remainder of the thesis we investigate more in depth the information associated with the referrer URL, with the purpose of presenting some applications that exploit its advantages.

Chapter 5

Estimating Page Importance on the BrowseGraph

In Chapter 4 we studied the informativeness of the referrer URL regarding the type of the navigation that the user will perform. We showed that the referrer might be correlated with the type of session of the user. In this chapter, we further analyze the user navigational patterns inside a website by studying the *BrowseGraph*, previously described in Section 3.3.2. This graph is built by aggregating the browsing sessions of the users: the nodes represent the web pages, and the edges represent the browsing transitions made by users.

Further in this thesis, we will exploit the collective browsing behaviors of users through the *BrowseGraph*, in order to personalize the content of the new users. Quantifying the importance of the nodes of this graph translates to identifying the most important content. Therefore, we perform a set of experiments in order to verify the performance of centrality-based algorithms, such as PageRank, on the *BrowseGraph*, extending the initial experiments performed by Liu *et al.* [72]. These experiments are performed on Yahoo News instead of Flickr, since we want to study the navigational patterns of users also on a different context. The results of this chapter are currently under review.

5.1. Introduction

The ability to identify the online resources that are perceived as important by users of a website is crucial for online service providers. Metrics to estimate the importance of the page from the structure of online links between them are widely used: algorithms that compute the *centrality* of the nodes in a network, such as PageRank [81], HITS [58] and SALSA [62], have been employed extensively in the last two decades in a vast variety of applications. Born and spread in conjunction with the growth of the Web, they can determine a value of importance of a page from the complex network of links that surrounds it.

More recently, centrality metrics have been applied to *browsing graphs*, (also referred to as *BrowseGraphs* [72, 100]) where nodes are webpages and edges represent the transitions made by the users who navigate the links between them. Differently from the hyperlinks network, this data source provides to the analyst a way to study directly the dynamics of the navigational patterns of users who consume online content. Also, unlike hyperlinks, browsing traces account for the variation of consumption patterns in time, for instance in the case of online news were articles tend to become rapidly stale. Comparative studies have shown that centrality-based algorithms applied over *BrowseGraphs* provide higher-quality rankings compared to standard hyperlink graphs [73, 72].

Most centrality measures aim at estimating the importance of a node, using information coming from the *global* knowledge of the graph topology—potentially the addition of new nodes and edges, can have a cascade effect on the centrality values of all other nodes in the network. This fact entails high computational and storage cost for big networks but, more critically, there are some situations in which a global computation on the entire graph is unfeasible, for example when the information about the entire network is unavailable. This is an important limitation in many real-world scenarios, where the graphs at hand are often very large (Web scale) and, most importantly, their topology is not fully known. This practical issue raises the problem of how well one can estimate the actual centrality value of a node by knowing only a local portion of the graph. This is known as the *Local Ranking Problem* (LRP) [29].

One of the questions behind LRP is whether it is possible to estimate efficiently the PageRank score of a web page using only a small subgraph of the entire Web [18]. In other words, if one starts from a small graph around a page of interest and extends it with external nodes and arcs (*i.e.*, those be-

longing to the whole graph), how fast will one observe the computed scores converging to the real values of PageRank?

We extend this line of work in the context of browsing graphs since they are often used and exploited in this thesis. We will estimate the importance of the web pages through centrality-based algorithms on the *BrowseGraph*, and the study presented in this chapter aims to verify the reliability of these approaches and their limitation on this different type of graph. We shed some light on the bias that PageRank incurs, when estimating the centrality score of nodes in a *BrowseGraph*, when only partial information about the graph is available. To do so, we monitor the browsing traffic of the news portal, and we extract different browsing subgraphs induced by the browsing traces of users coming from different *domains*, such as search engines and social networks. We consider the *BrowseGraph* built by the overall browsing patterns of users as the whole graph (*i.e.*, the Web graph for the hyperlink case), and the *subgraphs* extracted from it as the local graphs from which we aim to compute the centrality-based approaches. We learn that some strategies, of incremental addition of external nodes to these subgraphs, determine a fast convergence of the local PageRank to the global one.

We build on these findings with two different prediction experiments. For the first time we tackle the task of estimating *how much* the local PageRank diverges from the global one using only structural features of the local graph, usually available to the local service provider. Predicting the error of the local PageRank scores, in terms of distance from the global one, provides useful estimation to the provider about the reliability PageRank computed locally. Moreover, since in this dissertation we exploit the referrer URL for the purpose of ranking and recommendation, we need to face cases in which the domain of origin could not be readily available. This might happen for different reasons, when the users use third parties application such as clients of Facebook, Twitter and Mail, or when they rely on URL shortening services such as Bit.ly. As we showed in the previous chapters, the information about the domain from which the user is coming is valuable (especially in a cold-start case), as it gives an initial description of the user [32], and for this reason being able to infer it from the first browsing steps is important. Thus, we describe models that tell apart a subgraph from the others from just observing the behavior of a random surfer that navigates their links.

In summary, the main contributions of this chapter are the following:

- We study the *LRP* on a large-scale *BrowseGraph* built from a very popular news website, with the aim to validate the use of centrality-based approaches on this type of graphs. This chapter contributes to supporting the experiments on the Flickr *BrowseGraph* based on PageRank-like approaches, that we describe in Chapter 9. To the best of our knowledge we are the first to tackle this problem on the increasingly popular *BrowseGraph*.
- We show that an accurate estimation of the distance between the local and global PageRank, can be obtained looking at the structural properties of the local graph, such as degree distribution or assortativity.
- We tackle the problem of discovering the referrer domain of a user session, when this information is missing or hidden. We show that this is possible using a random surfer model, which is able to tell the referrer domain with high accuracy, just after the very first browsing transitions.

The remainder of the chapter is organized as follows. In Section 5.2 we describe our dataset and the extraction of the browsing graphs. In Section 5.3 we study the LRP problem on the *BrowseGraph* and compare the approximation accuracy of different graph expansion strategies. In Section 5.4 we present the prediction experiments. In Section 5.5 we wrap up and highlight possible extensions to the work.

5.2. Dataset

For the purpose of this study, we took a sample of user-anonymized log data of the Yahoo News network,[1] collected in 2013.

The data is comprised by a large number of pageviews, from which we extract the users' sessions as explained in Section 3.3.2. The final dataset contains approximately $3.8M$ unique pageviews and $1.88B$ user transitions.

In order to compare the behavior of different users that access the Yahoo News network from *different referrer URLs*, we build different *ReferrerGraphs* as explained in Section 3.3.2.

[1] We considered a number of different subdomains like *Yahoo news*, *finance*, *sports*, *movies*, *travel*, *celebrities*, *etc.*

Subgraphs	Nodes	Edges
Bing	61,531	255,464
Facebook	21,060	70,266
Google	142,646	779,185
Homepage	60,287	335,836
Reddit	2,445	4,868
Twitter	4,206	7,080
Yahoo	101,116	404,378

Table 5.1: Size of the extracted subgraphs.

On Table 5.1 a summary with the size of the graph, in terms of number of nodes and edges, is shown.

5.3. Analysis

5.3.1. Subgraphs Comparison

We extracted seven subgraphs from the main Yahoo News graph with the procedure discussed in Section 3.3.2. Since each subgraph is originated by sessions of users starting from the same domain, our hypothesis is that they somehow exhibit a different structure among them, and that on the contrary, the browsing patterns generated by different types of audiences might lead to different pieces of news pages, to emerge as the most central ones in the browsing graph. To check that, we ran the PageRank algorithm on each of the (weighted) subgraphs, and for every pair of subgraphs we compared the scores obtained on their common nodes using Kendall τ correlation between the rankings. The intersection between the node sets of the networks is always large enough to allow us to compute the distance on the intersection only (> 1000 nodes in the case with less overlap). Note that the intersection considers the nodes that are in common between the subgraphs, and this does not mean that the subgraphs have a similar structure in terms of edges (*i.e.*, users' traffic, and page importance). The main idea is to compute the local ranking of nodes on each subgraph: Kendall τ will provide a clear measure of how much the importance of the same set of nodes varies among different subgraphs. Therefore, if the ranking between two subgraphs differs

	Full	**Faceb.**	**Google**	**Bing**	**Yahoo**	**Reddit**	**Homep.**	**Twitter**
Full	1.0000	0.1791	0.3931	0.3278	0.3548	0.0656	0.2797	0.0764
Facebook	0.1791	1.0000	0.3146	0.4111	0.3430	0.2616	0.4070	0.3026
Google	0.3931	0.3146	1.0000	0.5815	0.5860	0.1088	0.4217	0.1297
Bing	0.3278	0.4111	0.5815	1.0000	0.6624	0.1469	0.5238	0.1688
Yahoo	0.3548	0.3430	0.5860	0.6624	1.0000	0.1245	0.4632	0.1386
Reddit	0.0656	0.2616	0.1088	0.1469	0.1245	1.0000	0.1534	0.2309
Homepage	0.2797	0.4070	0.4217	0.5238	0.4632	0.1534	1.0000	0.1523
Twitter	0.0764	0.3026	0.1297	0.1688	0.1386	0.2309	0.1523	1.0000

Table 5.2: Kendall τ correlations between PageRank values ($\alpha = 0.85$).

greatly (*i.e.*, it has a very low Kendall τ), this fact can be interpreted as an indication that they either show different content, *i.e.*, web pages, or anyway that their content has a very different order or importance.

Table 5.2 reports on the cross-distance among the subgraphs and also with respect to the full graph using Kendall τ. Interestingly, most of the similarity values tend to be very low (lower than 0.3), confirming the hypothesis that the user's interests are tightly related to the domain where they come from. Some of these similarities, however, are considerably higher, remarkably the ones between the three subgraphs that are originated from search engines traffic, *i.e.*, Bing, Google and Yahoo, which yield the most similar rankings of pages (greater than 0.5).

5.3.2. "Growing Balls"

We first focus on the study of the *Local Ranking Problem* on browsing graphs. An important question related to this problem is how much the PageRank node values vary when new nodes and edges are added to the local graph. A natural way to determine this is to expand incrementally the graph by adding new nodes and edges in a Breadth-First Search (BFS) fashion and comparing the PageRank computed on the expanded graph with the one on the global graph.

More formally, given a graph H which is a subgraph of the full graph G, we simulate a growth sequence H_0, $H_1 \dots H_n$ in the following way:

- $H_0 \longleftarrow H$;
- $V_{H_{k+1}} \longleftarrow \{\Gamma_{out}(V_H) \cup V_H\}$, with V_x being the set of vertices of a graph and Γ being the vertex neighborhood function;
- $E_{H_{k+1}} \longleftarrow \{(v_1, v_2) | v1 \in V_{H_{k+1}} \wedge v2 \in V_{H_{k+1}}\}$, with E_x being the set of edges of a graph.

Using the standard graph terminology, we refer to the various steps of this expansion as *"balls"*, where the ball H_0 is the initial subgraph and subsequent balls are obtained by adding all the outgoing arcs that depart from the nodes in the current ball and end in nodes that are not in the set of nodes of that ball. Observe that, depending on how it is built, H_0 may not be an induced subgraph of G, but $H_1, \cdots, H_n$ are always induced by the expansion algorithm.

Using the Kendall τ correlation we measure the difference between the local PageRank computed for each ball H_i and the global PageRank computed on G. The main objective is to understand how much the ranking gets close to the global one at each consecutive step, and whether the ranking values are able to converge even if we just consider a piece of the information contained in the whole graph.

To check the dependency of results we consider three different sets of initial subgraphs, that we will study separately. It is important to experiment with different types of subgraph, for example, in order to understand whether the results are related to the type of nodes in each graph (*e.g.*, different pages visited) or rather to the structure of the same (*e.g.*, by their size). We describe them next:

- ***ReferrerGraph* (RG)**. The seven browsing subgraphs built by referrer URL: Facebook, Twitter, Reddit, Homepage, Yahoo, Google and Bing;
- **Same size *ReferrerGraph* (SRG)**. To measure how much the different sizes of the graphs impact the observed behavior, we fix a number of nodes and extract a portion of each subgraph in order to obtain exactly the same size. The selection is performed with a sequence of

BFS expansions, starting from a random node in each graph. Then, we select a resulting subgraph for each graph so that they have a very similar size (±9.4%): other ways of selecting subgraphs would end up with disconnected samples, which of course would void the purpose of this experiment. Doing so, we are able to compare the graphs on equal grounds and at the same time control for the effect of the size (about $3K$ nodes and $20K$ edges).

- **Random (R)**. To check whether the observed behavior has to do with the user behavior underlying the graph under examination (*e.g.*, the particular structure of the graph determined by the sessions of users coming from Twitter), we take a set of seven *random* graphs from the whole *BrowseGraph* (full) without considering the referrer, where each of them reflects the size of each of the referrer-based subgraphs. In this way we can explore the behavior of browsing graphs that preserve the size of the graphs originated by specific types of users, but that are "artificial" in the sense that destroy any connection with the behavior connected to users coming from the same domain. To make sure that the size is the same we start from a BFS exploration and we prune the last level to match exactly the size we need.

The results of the "Growing Ball" experiment applied to the **RG** case are shown in Figure 5.1 (top). The y-axis shows the Kendall τ correlation between each subgraph at the step x and the full graph (*BrowseGraph*). The convergence happens relatively quickly, as the correlation τ approaches 1 in the first 3 iterations. The curves related to different subgraphs are shifted with respect to each other, apparently mainly due to their different size, the biggest networks starting from higher τ values and converging faster than the smaller ones. To determine the dependency on the graph size, we repeat the same experiment for the **SRG** case. The results for this case are shown in Figure 5.1 (center). Even if the curves resulted to be more flat (confirming that the initial size has indeed a role in the convergence), we still observe noticeable differences between the curves for the first two expansion levels, meaning that different subgraphs are substantially different from one another in terms of their structure: even after forcing them to have the same size, the convergence rates observed on the different graphs varies. At the first iteration, for instance, all the subgraphs in **SRG** have Kendall τ between 0.3 and 0.5, whereas the ones in **RG** are between 0.4 and 0.6. Moreover in **SRG** the biggest networks starting from higher τ values do not converge faster. This intuition is confirmed by repeating the

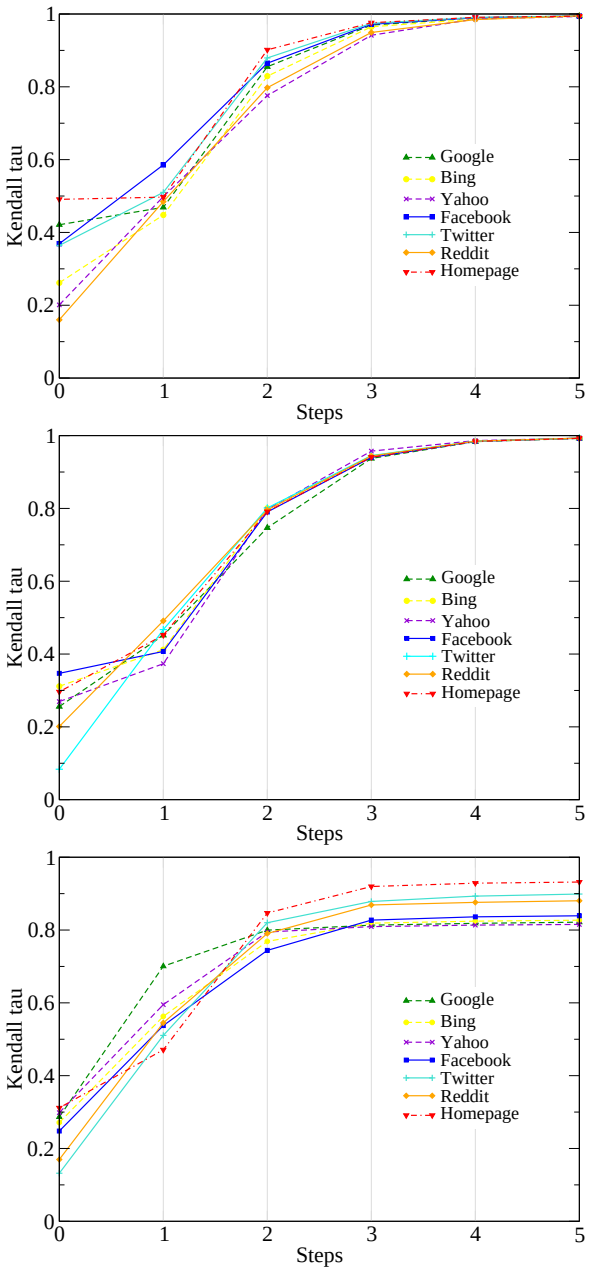


Figure 5.1: Growing Balls experiment on: original subgraphs built based on the referrer URL (top), seven subsubgraphs with very similar size (center), seven subgraphs random selected from the full graph (bottom), where each of them has the same size of one of the original.

experiment on graphs selected with the **R** strategy. Results, displayed in Figure 5.1 (bottom), show that convergence in this case is much slower and the difference between the curves is less prominent. With the previous experiment we show that the Growing Balls on random subgraphs behave differently, especially when considering the number of iterations required in order to converge.

5.3.3. Growing Balls with Selection of Nodes

Besides the selection of the initial graph, the rank convergence depends also on the way the growing balls are built at each iteration. How does the expansion influence convergence if only a few more representative nodes are selected? To what extent a higher *volume* of selected nodes helps a quicker convergence or adds more *noise*? At a first glance, one may argue that using all the nodes means injecting all the available information, so the convergence to the values of PageRank computed on the full graph G should be faster. On the other hand, instead, one may observe that we are introducing a huge number of nodes in each iteration (as the growth is at each step larger) adding also the ones that are less important and this can induce an incorrect PageRank for some time, until all the graph becomes known. In order to shed light on this aspect, we introduce a variant in the growing-balls expansion algorithm and we select only the nodes with highest PageRank.

More formally, considering H_k as the subgraph at iteration k and V_{H_k} its set of nodes, we select all the external nodes in $Y = \{V_G\ V_{H_k}\}$, that are connected through outgoing arcs from the nodes in V_{H_k}. We then compute the PageRank values on the subgraph H_k extended with the nodes Y, and obtain a ranked list of nodes. Among all the nodes in Y we select the top $n\%$ with largest PageRank value, and only those ones will be added to H_k in order to build H_{k+1} and advance to the next iteration.

We conducted experiments with this partial expansion at different percentages of neighbors selected at each step: 5%, 10%, 30%, 50%, and 100%, and then we computed the average Kendall τ value for each one of the percentages. The results for some representative cases are shown in Figure 5.2. The partial expansions of 20% and 30% are not shown in order to improve the clarity of the plot, since they do not add any additional information. Remarkably, the figure highlights how expanding the graph by adding fewer nodes, although the most representative ones, leads to PageRank values that are closer to the *global* ones in the first iterations. Since we are expanding the

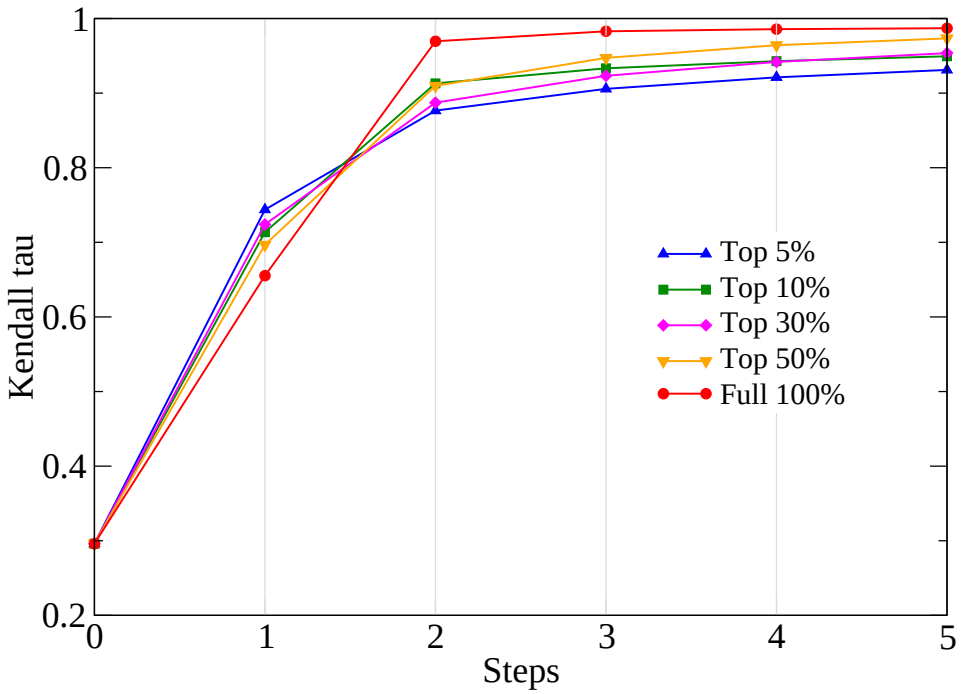


Figure 5.2: Growing Balls using only the nodes with highest PageRank. The plot shows the average values of the Kendall-τ at each step computed for all the subgraph.

local graph with a small (highly-central) number of nodes, we could argue that they initially help to boost the local PageRank scores. However, given that we keep on expanding using a few nodes at each iteration, the nodes that have not been added before exclude a large number of nodes among which there might also be highly central ones. This might explain why in the first iteration(s) the convergence rate is high, but on the limit the final convergence values result in a low Kendall τ. Contrarily, in the long run, expansions that include the highest number of nodes present convergence values closer to 1. This is somehow expected, given that at each iteration any subgraph H closer in size to the full graph G, will include almost every node and arc.

Nonetheless, the main significant outcome of this experiment is that it is possible to obtain a yet satisfactory PageRank convergence with few but very representative nodes. Considering situations in which including additional pieces of information, in terms of node/arc insertions, implies a non-

negligible cost; requesting just a little amount of well-selected information allows to obtain good approximations while minimizing the costs.

5.4. Prediction

In the previous section we showed how the approximation to the global PageRank varies with the expansion of the initial subgraph. The ranking of the nodes converges quite fast on all the subgraphs: they differ in terms of their content, although they are similar in terms of structure in that all of them are built based on users' navigational patterns. Building upon the findings about how local and global PageRank computed on the *BrowseGraphs* relate to each other, we designed two different experiments to assess how well a learned model could perform in predicting this relationship.

First, we simulate a user behaving as a random surfer who is navigating the links of a specific referrer-based graph, with the task of identifying which graph it is among all the referrer-based graphs we introduced, by observing the smallest possible number of page transitions of the surfer. It is important to remark that all the subgraphs are extracted from the same larger browse graph and they could share some nodes and arcs, more in general their structure, because all of them are built based on the users browsing sessions. The main research question we address is if we can easily identify in which subgraph the surfer is browsing, and how many iterations we need in order to predict the user's original subgraph within a certain accuracy threshold.

Second, we address the problem of predicting the Kendall τ between the local and the global PageRank, only considering information available on the local graph such as topological features. This is an extremely common situation given that, in general, the information pertaining to the local graph is the only one that is readily available, and usually of a limited size so it is quite feasible to work with it in terms of computational cost.

5.4.1. Random Surfer

In Section 5.3 we showed how users coming from different sources behave differently in terms of content discovery, in addition to what we showed in Chapter 4 about the relation between the referrer and the type of session made by the user. Being able to understand the browsing trail followed by a particular user is a strong signal of the user's behavior, allows for a greater level of customization of web pages to the user's interests, and for tailored advertisements. However, the information of the user's referrer URL

Algorithm 1: `RandomSurfer`(k, α, steps, G)

```
logPr ← initialize vector with size G_k.length();
n ← total number of nodes;
x_j ← choose (random) starting node ∈ G_k;
/* For each step, compute a random walk in G_k, and compare the
probability to be in all the other G */
for s ← 1 to steps do
    /* Pick the next node of G_k with random walk */
    x_k = next_node( G_k, x_j );
    for i ← 0 to G.length() do
        ⟨k_out⟩ ← get_outdegree(n_p);
        if ⟨k_out⟩ == 0 then
            logPr[ i ] ← logPr[ i ] + log(1/n);
        else
            p_i(x) = (1 − α)/n;
            Pd_{x_j} ← get_probability_distribution(G_i, x_j);
            S_{x_j} ← get_successors(G_i, x_j);
            if x_k ∈ S_{x_j} then
                p_i(x) ← p_i(x) + α ∗ Pd_{x_j}(x_k);
            logPr[ i ] ← logPr[ i ] + log(p_i(x));
return logPr
```

is not always visible and, in many cases, it is hidden or masked by services or clients. For instance, any Twitter or mail client (*i.e.*, third-party application) shows an empty referrer URL in the web logs; a similar situation happens with the widespread URL-shortening services (*e.g.*, `Bitly.com`), that mask the original web page the user comes from. Nonetheless, in all these cases, a provider could make use of her knowledge of the user's trail to automatically identify the source where the user started her navigation in the local graph. As we have shown, the referrer URL might be useful to characterize the interest of the users, especially in the case where the users are unknown (*i.e.*, the user profile is not available). Thus, being able to identify the referrer URL when it is not available, translates into an advantage for the content provider.

Therefore, we decided to consider the following scenario: a content provider

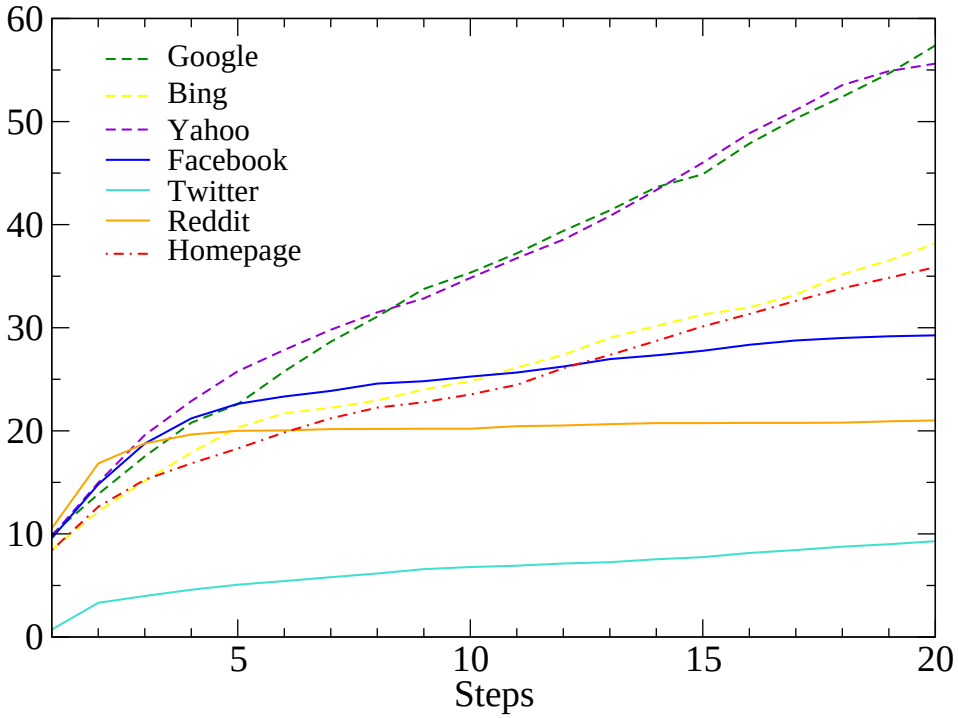


Figure 5.3: Random Surfer Experiment. On the y-axis: log-ratio of the probabilities (as explained in the text). X-axis: number of browsing steps performed by the surfer.

is observing a user surfing the pages of its web service, but she is unaware of the user's referrer URL. In terms of our experimental dataset, this scenario maps into the problem of observing a browsing trace left by a random surfer, on one of the referrer-based subgraphs, and having to identify which graph it is. Intuitively, the larger the number of page visits (or *steps*) the surfer will make, the more distinctive its trace will be, and the easier the identification of the graph. Algorithm 1 shows the pseudo-code that describes the process to compute the random surfer experiment.

Formally, observing the sequence of visited nodes $\mathbf{x} = (x_1, x_2, \ldots, x_s)$ and computing the probability $p_i(\mathbf{x})$ that the surfer has gone through them, given that it is surfing G_i, we need to deduce what is G_i, *e.g.*, by maximum log-likelihood. With this aim in mind, we sort the indices of the subgraphs $i_1, i_2, \ldots$ so that $p_{i_1}(\mathbf{x}) \geq p_{i_2}(\mathbf{x}) \geq \ldots$ and stop as soon as the gap between $\log p_{i_1}(\mathbf{x})$ and $\log p_{i_2}(\mathbf{x})$ is large enough.

In this set of experiments, we considered the seven URL-referral subgraphs $G_1, \ldots, G_7$, one at a time. For each subgraph G_i, we simulated a random surfer moving around in G_i (*i.e.*, calling the function `RandomSurfer`(i, α, steps, G)), computing at each step (*i.e.*, page visited) the probability of the surfer to navigate in each subgraph $G_1, \ldots G_7$: we expect that the probability corresponding to G_i will increase at each step, and will eventually dominate all the others.

In order to estimate the number of steps required to identify correctly the graph that the surfer is browsing, we measure the difference between log-probabilities for the correct graph G_i, and for the graph with the largest log-probability among the other ones. As with PageRank, we introduced a certain damping factor ($\alpha = 0.85$); this is necessary to avoid being stuck in terminal components of the graph. Recall that α, is the balancing parameter that determines the probability of following the random walk, instead of teleporting [81]. The results are shown in Figure 5.3, averaged over 100 executions. The values on the y-axis represent the difference between the log-probabilities (*i.e.*, the logarithm of their ratio): in general, we can observe that the very firsts steps are enough to understand correctly (and with a huge margin) in which graph the surfer is moving. The inset of Figure 5.3 displays the first 20 steps and the relative probability to identify the correct graph. Almost all the referrer domains are recognizable at the first step. This translates in a strong advantage for the service provider, as it can identify from where the users are coming from, even if they use clients or services that masquerade it. With this information the service provider can personalize the content of the web pages for any users with respect to the referrer.

Interestingly, the plot reveals a different fact, namely that some surfers are easier to single out than others; we read this as yet another confirmation that the subgraphs have a distinguished structural difference, or (if you prefer) that users have a markedly different behavior depending on where they come from. However, our experiment shows that it is possible to identify the referrer of the user's session, even when this is not available in the browsing log. This allows to use the methods and algorithms based on the *ReferrerGraphs* discussed in this thesis, also when that information is initially missing.

5.4.2. Prediction of Kendall Tau Correlation

We have seen that the deviation of the local PageRank, with respect to the global one can be relevant, depending on factors such as the size of the local graph and the different behavior of the users who browse it (see Section 5.3.2 and particularly Figure 5.1). Recall that we compute the distance comparing the rankings with the Kendall τ, since we are interested in obtaining a ranking as close as possible to the one computed on the entire graph. Although we have previously shown how to expand the view on the local graphs with nodes residing at the border, this practice might not always be possible in a real-world case, since service providers often can have access only to the browsing data generated in their servers.

Previous work on local ranking on graphs raised several questions related to this scenario, highlighting practical applications of the local rank estimation non only for web pages but also in social networks [18]. Critically, so far it is not clear whether there are some topological properties of the local graph that make the local ranking problem easier or harder, and if these properties can be exploited by local algorithms to improve the quality of the local ranking. We explore this research direction, by studying a fundamental aspect that is at the base of the open questions in this area, namely the possibility of estimating the deviation of the local PageRank from the global one, using the structural information of the local network. The intuition is that, some structural properties of the graph could be good proxies for the τ correlations, computed between local and global ranks. Being able to estimate the Kendall τ distance between the subgraph available to the service provider and the global graph, implies the ability to estimate the reliability of the current ranking using only information of the local subgraph.

To verify this hypothesis we resort to regression analysis. Starting from the seven subgraphs in the dataset, we build a training set using the jackknife resampling technique, by removing nodes in bulks (1%, 5%, 10%, 20%) and computing the τ value between the full subgraph and their reduced versions. Then, for each instance in the training set, we compute 62 structural graph metrics [114, 9] belonging to the following categories:

- **Size and connectivity**. Statistics on the size and basic wiring properties, such as number of nodes and edges, graph density, reciprocity, number of connected components, relative size of the biggest component.

- **Assortativity**. The tendency of a node with a certain degree, to be linked with nodes with similar degree. We computed different combinations that take into account the in/out/full degree of the target node vs. the in/out/full degree of the nodes that are connected with it.
- **Degree**. Statistics (average, median, standard deviation, *etc.*) on the degree distribution of nodes.
- **Weighted degree**. Same as **degree**, but considering the weight on edges, that usually referred to as node strength. As the edges are the transitions made by users during the navigation, the weight stand for the number of times the users have navigated the transition.
- **Local Pagerank**. Statistics on the distribution of the PageRank values computed on the local graph.
- **Closeness centralization**. Statistics on the distances (number of hops), that separate a node to the others in the graph, in the spirit of the closeness centralization [114].

We employed different regression algorithms, although we report the performance using random forests [16], which performed better in this scenario than other approaches like support vector regression [94]. We computed the mean square error (MSE) across all examples in all sampled subgraphs. The mean square residuals, obtained over a five-fold cross validation computed on a random forest regression, is very low, around $2.4 \cdot 10^{-6}$. Results, computed for the full set of features and for each category separately, are given in Table 5.3. The most predictive feature category is the weighted degree, which yields a performance that is better (or comparable) than the model using all the features. This might be due to the fact the model with 62 features is too complex for the amount of training data available. On the other hand, the *assortativity* features seem to be the ones that have the least predictive power on their own.

We then use the learned model to predict the τ values of the seven subgraphs. When we applied the predictive models learned in the subsamples to regressing the full graphs, the MSE is less than 0.026 on average, which, even if relatively low, is higher than the cross-validated performance in the sub-samples. However, the model was able to rank the seven different subgraphs by their Kendall τ almost perfectly. When using all the features the

Feature Class	**MSE**
weighted degree	$2.2 \cdot 10^{-6}$
size and connectivity	$2.7 \cdot 10^{-6}$
degree	$3.3 \cdot 10^{-6}$
closeness	$4.2 \cdot 10^{-6}$
local PageRank	$4.6 \cdot 10^{-6}$
assortativity	$9.0 \cdot 10^{-6}$
ALL features	$2.4 \cdot 10^{-6}$

Table 5.3: MSE of cross validation. Average differences are statistically significant with respect to *weighted degree* and *ALL features*, (t-test, $p<0.01$).

Spearman's correlation coefficient between the true order and the predicted one is 0.85 (high correlation), and when we used the most predictive features (weighted degree) the correlation was as high as 0.80 (moderate high correlation). Overall, the final rankings are just one swap away (Kendall's τ is over 0.70 in this case).

This kind of information can be very helpful when comparing different local sub-domains to determine which one has pages that better estimate the global PageRank.

5.5. Summary and Discussion

In this chapter we analyzed different aspects regarding the *BrowseGraph*. We studied how a centrality-based algorithm performs on this type of graph, in order to estimate the importance of the web pages. In particular, we tackled the *Local Ranking Problem*, *i.e.*, how to estimate the PageRank values of nodes when a portion of the graph is not available, which arises commonly in real use cases of random walk approaches. We investigated this problem for the first time in a novel environment such as a large user-generated browsing graph from Yahoo News.

We built different *ReferrerGraphs* and studied the different web pages (*i.e.*, nodes) consumed among the various subgraphs. Interestingly, the browsing patterns initiated from different domains exhibit remarkable differences in terms of which pages users visited next. In practice, this observation implies

that users' interests could be partially modeled by knowing where the users are coming from, therefore, opening possibilities for personalization and page content-optimization services. In Chapter 7 we will discuss a recommender system based on these *ReferrerGraphs*.

With this observation in mind we performed several experiments using a very large network of sites, with almost two billion user transitions. We assessed to what extent the information of the browsing patterns can be generalized when only the information from smaller subgraphs is provided.

First, we computed the PageRank of the subgraphs on their step-by-step BFS expansion, and we measured the correlation of, in terms of Kendall τ, with the PageRank computed on the full graph. To control the subgraph size and type, and to study the impact of the expansion strategy on the PageRank convergence, we used two flavors of BFS and three different sets of initial subgraphs.

We found that expanding the local graph with a few nodes of the largest value of PageRank leads to a faster (although less accurate, in the long run) convergence. On the other hand, adding more nodes leads to a slower convergence rate in the first steps. Therefore, in all the cases where a strong convergence with the values of the global PageRank is not required, selecting few specific nodes is enough to significantly improve the PageRank values of the local nodes, without requesting and processing a huge volume of data.

We also performed a series of experiments with the aim of predicting which referrer URL the user joined the network from, *i.e.*, if a model can predict reliably where the user is entering our network. In general, just after few steps (*i.e.*, few visited pages), it is already possible to recognize the referrer URL correctly—the surfing behavior is very distinctive of the domain the user is coming from. With this approach it is possible to allow early user personalization also in all those cases where the domain from where the user is coming from is not available, such as Facebook, Twitter clients, URL shortening services, and so on.

Finally, we performed another experiment trying to predict the value of the Kendall τ between the local and the global PageRank only considering information available on the local graph. We explored six different sets of topological and structural features of the browse graph, namely size and connectivity, assortativity, degree, weighted degree, local PageRank and closeness. We computed those features on a training set we obtained by applying a jackknife sampling of our subgraphs and we ran a regression model on

the Kendall τ of the PageRank of the full subgraph and the various samplings. We found that a random forest ensemble built on *weighted degree* outperforms all the other features in terms of mean square error. When applying the regression model to the task of predicting the τ value of the global graph with the seven subgraphs at hand, we were able to reproduce quite well the ranking of their estimated τ values with their actual ranking, up to a Spearman's coefficient of 0.8.

The findings observed in this chapter serve as guides for a set of recommender systems proposed and discussed in Chapter 7. In that chapter we will analyze the user behavior in the news context more in depth, and we will propose and compare 24 flavors of recommender systems. Moreover, the experiments related to the Local Ranking Problem support the use of PageRank and other centrality-based algorithms that we discuss in Chapter 9 for ranking of items.

Part II

Implicit Information in Recommendation

Chapter 6

Recommendation of Photostreams

In the third part of this thesis, we investigate recommendation approaches that are based on user browsing data. We present a collaborative filtering recommender that exploits user navigational patterns, in order to recommend the next item to the user that is currently browsing the website. In particular, we focus our analysis and experiments on Flickr, addressing a novel recommender problem where the items are represented by entire photo albums instead of individual images. In photo-sharing websites and in social networks, photographs are most often browsed as a sequence: users who view a photo are likely to click on those that follow. The sequences of photos, which we call *photostreams*, as opposed to individual images, can therefore be considered to be very important content units in their own right. In spite of their importance, those sequences have received little attention, even though they are at the core of how people consume image content.

In this chapter, we focus on photostreams, first performing an analysis of a large dataset of user logs, examining navigational patterns between photostreams. Then, we implement two stream recommendation algorithms and we evaluate them through a user study. Our analysis yields interesting insights into how people navigate between photostreams, while the results of the user study provide useful feedback for evaluating the performance and characteristics of the recommendation algorithms. The results of this chapter were published in [31].

6.1. Introduction

Social media platforms, such as Flickr provide a wide range of functionalities and different ways to share and view content. Given the sequential nature of browsing photographs, it is common for people to share and view images in sequences, whether the photos are arranged in galleries, slideshows, or in groups. Particularly in Flickr, photos uploaded by a user to his account are placed in a "photostream", which in essence is a sequence of photos. Although there are many ways to reach individual photographs, such sequences constitute a fundamental part of the interaction. In the rest of the chapter, we will refer to such sequences as *photostreams*, or simply *streams*.

Navigation across sequential units of content is also present in other fields of social media, *e.g.*, social networks, music streaming and microblogging platforms. In popular social networks, photos are organized in albums and can be viewed sequentially. Songs in music streaming services can be listened to one after another, usually as a part of an album or a playlist. Posts in microblogging platforms are chronologically organized in independent blogs. Therefore, methods developed for photostreams could be adapted to other social media as well.

A key question regarding photo-sharing platforms is then, "how users navigate inside and between various photostreams". In particular, such photostreams may be considered not just collections of images, but rather fundamental units of content. On one hand, understanding how users navigate between specific photostreams is crucial in designing interfaces and algorithms that improve user experience, by providing the right content in the right places. On the other hand, analyzing the semantic categories of such streams can also provide important insights on general topics of interest. In addition, investigating the transition of users between photostreams allows us to understand how topics may be related.

In this chapter, we treat photostreams as individual units of content, and analyze a large sample of navigation logs to gain insights into how users navigate between different photostreams. More specifically, we examine user navigation logs containing several million pageviews, in order to create a photostream transition graph to analyze frequent topic transitions (*e.g.*, users often view "train" followed by "firetruck" photostreams). We implement two photostream recommender systems: a collaborative filtering approach based on transitions between photostreams, and a content-based method using tag-similarity of photostreams. Finally, we report the results of a user

study involving 40 participants to explore the fundamentals for the design of an effective recommender system in a large social media platform.

The main contributions of this chapter can be summarized as follows:

- We perform a large-scale analysis of photostream browsing patterns, providing insights into frequent transitions between different topics in image browsing.
- We propose a collaborative filtering recommender system based on historical users' browsing patterns in order to recommend photostreams, and we compare it with a standard content-based recommender.
- By means of a user study, we show that the collaborative filtering method, based on transitions between photostreams, provides more novel content than the tag-based recommender system.

To the best of our our knowledge, this is the first study which analyses photostream browsing as opposed to individual photo browsing. This is also the first time the problem of photostream recommendation is addressed, in particular by leveraging the navigation patterns of a large number of users.

6.2. Dataset

For the purpose of this study, we took a sample of the pageviews of more than 10 million anonymous Flickr users from 2011. The details of the dataset, the data selection and the data filtering can be found in Section 3.3.1.

Tags of Photos

Users in Flickr can create and attach tags to their photos in order to organize them and increase they reachability in the social network. We gathered tags of all public photos in the dataset from Flickr. We pre-process these tags by discarding the ones that belong to a multi-lingual stop-word dictionary, obtaining around 5 million distinct tags. We use these tags in order to compute the similarity among the photostreams.

6.3. Analysis

In this section we define the main concepts of our study, present statistics on how users browse within sessions, and on how the transitions between photostreams occur.

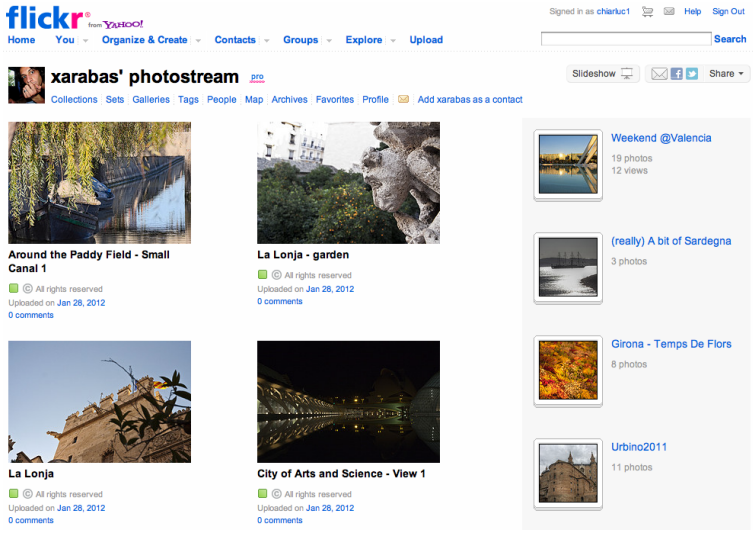
6.3.1. Photostream Browsing

Photos in Flickr are organized in photostreams. Each photo in Flickr belongs to a photostream of the owner, but it can belong to other streams of photos as well: groups, sets, galleries, favorites, *etc.* Apart from favorites, all of these photostreams are either chosen or created by the owner of the photo. Users always view and browse photos in the context of a particular photostream.

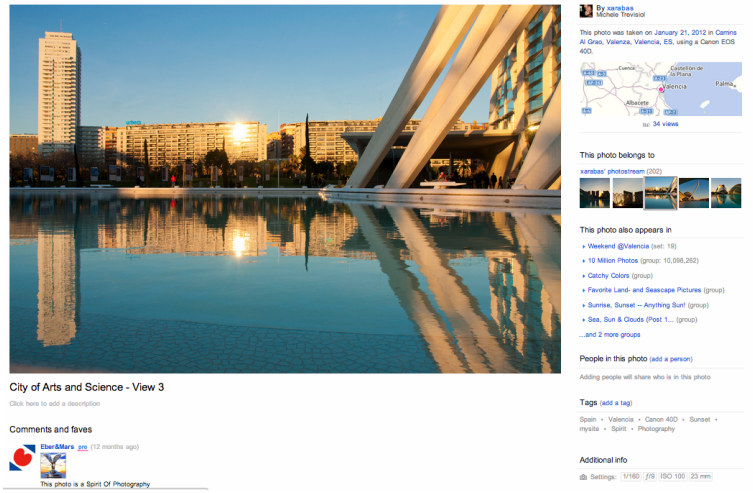
There are two main ways of viewing photostreams: *a*) *grid view*, *i.e.*, grid of photos from the stream (see Figure 6.1a), and *b*) *photo-focused view i.e.*, a single zoomed-in photo with a possibility of browsing neighboring photos (see Figure 6.1b). Although Flickr allows different variations of grid views, they share a common feature, namely that they show several pictures from the browsed stream at a glance. The photo-focused view is the same for all the streams: it shows a large selected photo and, on the right side of it, thumbnails of 4 neighboring photos from the stream are presented, which the user can switch to by clicking on them. This way one can change the focus from the current photo to another one from the currently browsed stream. A list of all photostreams that the photo belongs to is shown below the thumbnails in the form of hyperlinks, as visible in Figure 6.1b.

One can expect that users first enter the grid view of a photostream, and then select one of the photos they like and see it in a photo-focused view. Then, they can continue on browsing other photos from this photostream. The grid view may be used for purposes which seem less involving to the user, *e.g.*, quick browsing many photos from a stream, having an overview of a stream or seeking interesting content. Photo-focused views give the user options of performing many different actions in reference to the photo, *e.g.*, he or she can comment on the photo, favorite it, download it, see it in different sizes or in a light-box setting.

For the purpose of the study, we define a *stream-browsing sequence* as an uninterrupted chronological sequence of pageviews, that contains at least one photo-focused view and an indefinite number of grid views of one particular photostream (schematic examples are shown in Figure 6.2). Each browsing session can consist of a number of stream-browsing sequences.



(a) grid view



(b) photo-focused view

Figure 6.1: Two different types of stream views in Flickr. The first one shows a grid of small images for the photos that belong to that photostream, the second one allows instead to slide one image at a time.

The Flickr log sample in our dataset, contains a total of 264 million pageviews, out of which a considerable part form stream-browsing sequences. On average, each sequence consists of 8 pageviews, among which there are photo-

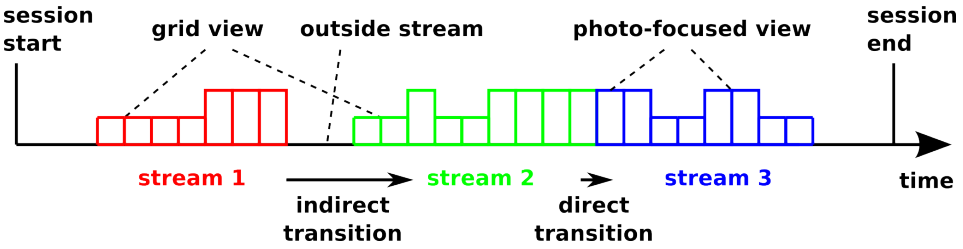


Figure 6.2: Diagram of possible transitions between streams.

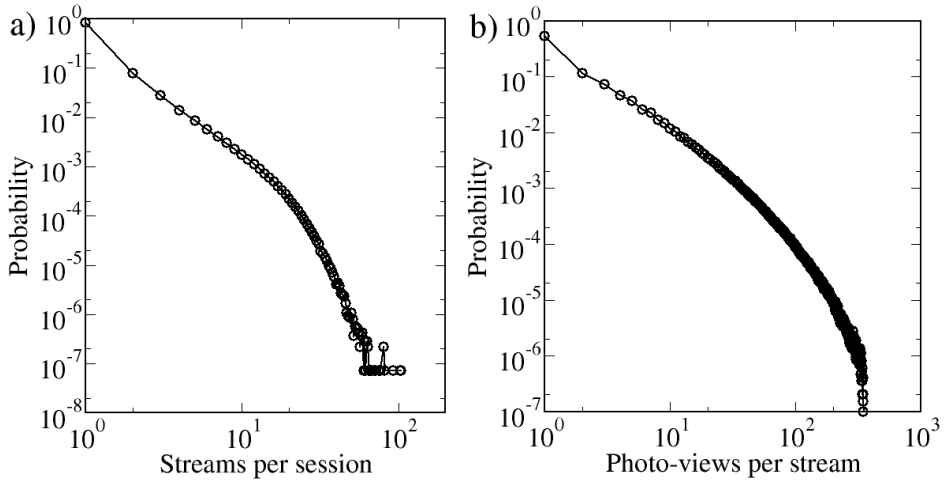


Figure 6.3: Distributions of number of unique streams per session (a) and number (log-log scale) of photo-focused views per each unique stream in a session (b).

focused views and grid views of the photostream. Distributions of both the number of distinct streams viewed per session (Figure 6.3a), and the number of photo-focused views per stream (Figure 6.3b), have a heavy-tail showing high variability in user browsing patterns.

6.3.2. Transitions Between Streams

In the previous section, we showed that a large portion of all pageviews corresponds to sequential browsing of photos inside photostreams. In this context, an interesting question to ask is how users switch between streams.

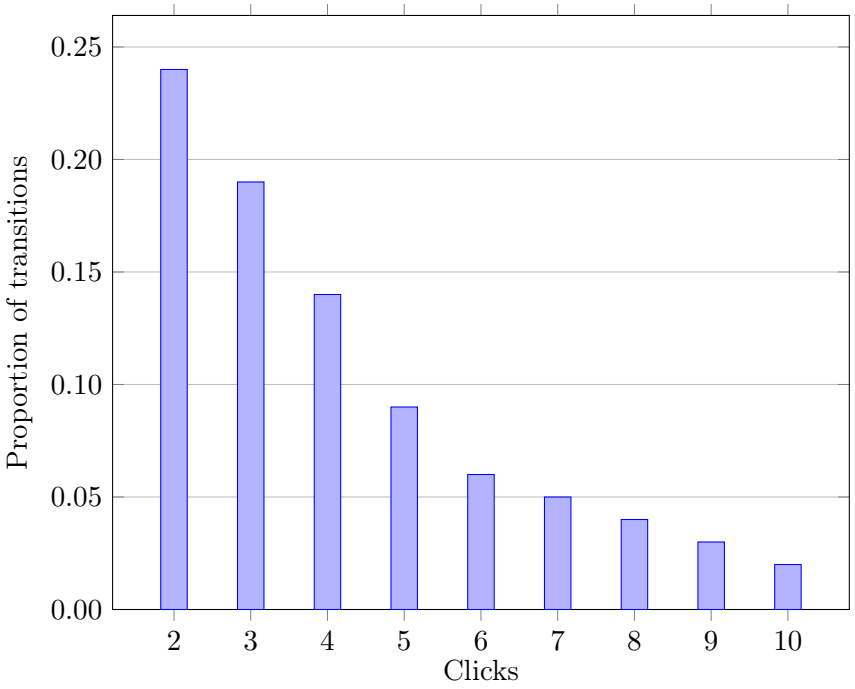


Figure 6.4: The number of clicks between different streams is shown.

We distinguish two types of transitions, that are shown in Figure 6.2: *a*) *direct transitions*, which happen when the user is in photo-focused view of the stream and chooses one of the listed photos inside the same stream, that are recommended to the right of the photo as shown in Figure 6.1b, and *b*) *indirect transitions*, in which the user leaves the photo-focused view and enters it again in a different stream after performing a number of clicks visualizing different content and interfaces, for example, watching grid views, searching, exploring users' profiles, *etc.*

We define a transition from photostream i to j, as a sequence of non-photo-focused pageviews from a photo-focused view inside stream i to another photo-focused view inside stream j. This definition implies directionality. One can estimate the number of clicks and actions performed during the transition, by counting the number of pageviews between the photo-focused views of the two streams and summing one. Direct transitions only require one click, whereas indirect transitions require more than one action.

In total, we identified 3.6 million transitions between photostreams. Indirect transitions, achieved within 2 clicks cover a large portion of all transitions,

as shown in Figure 6.3c. However, even more transitions happen after more than 5 clicks, so many users, before reaching another picture in a photostream, pass through many non photo-focused pageviews. Moreover, direct transitions happen much less often than indirect transitions.

Discussion of the Analysis

Users tend to see multiple photos of a photostream either in the photo-focused view or in the grid view before leaving the stream. The vast majority of all transitions between photostreams take place over several clicks. These results suggest that a modified photo-focused interface that facilitates direct transitions to other streams could be implemented. Moreover, a system recommending other photostreams within this interface could be an improvement.

6.4. Recommendation of Photostreams

In this section we introduce two recommender systems that suggest photostreams (and the photos belonging to them). The first is based on collaborative-filtering, specifically, on the transitions between photostreams from past user browsing sessions. The second is based on the content, *i.e.*, the tags of the images in the photostreams. Our aim is to compare these two approaches in terms of interestingness, relatedness and novelty of the recommended photos. Although we use well-known recommendation techniques, the novelty of our system lies in the use of photostreams as the main content unit. The systems consist of two levels. First, we recommend photostreams; second, we center on related photos from the photostreams. In this section, we describe the recommender systems in detail, while in the following section we evaluate them with a user study.

6.4.1. Two-Level Recommender System

In this section, we propose two distinct recommender systems based on anonymized traffic data and content data. As such they do not require the user to log in, therefore, these recommender systems are suitable also for newcomers. Note that, all the approaches presented in this thesis that are based on user implicit actions, are suitable for cases of cold-start. Each recommender system is built with a top-down approach in mind, meaning that it first analyzes high-level content units (photostreams) and then low-level content units (photos). Both the recommenders consist of two levels:

1. *Photostream selection*: the system recommends a set of streams to the user based on the streams the user has seen until that moment.

2. *Centering on a photo inside a photostream*: the algorithm chooses which part of the stream (*i.e.*, consecutive photos) will be displayed to the user for each selected stream based on the last seen images. Recall that there are 5 photos from the stream that are selected as representatives, as shown in the green box in Figure 6.5a.

Due to the fact that both levels function on significantly less data (photostreams), this two-level system is computationally much less demanding than a system working at the level of all the single photos. For example, the first level effectively reduces the task of giving recommendations among billions of photos, to the task of providing recommendations among just millions of streams. Moreover, the two-level design lets us circumvent the problem of data sparsity inherent in highly atomized social media platforms, which commonly store billions of images.

6.4.2. Photostream Selection

Here, we describe the first level of the two photostream recommendation algorithms. The task at this level is to rank photostreams based on the browsing history of the current user.

Collaborative Filtering Recommender

The first recommender presents to the user those streams which were most often co-viewed in the past sessions with the streams seen by the user in the current session. The algorithm computes the relevance of unseen streams in the following way:

1. Given a stream i in the current session and an unseen stream j, we compute c_{ij}, *i.e.*, the number of past sessions in which they appear together.

2. Then, we consider the last $N_s = 5$ streams from the current session, and for each unseen stream j we compute its relevance to the current session: $c_j = \sum_{i=1}^{N_s} c_{ij}$.

3. Finally, the recommender selects the streams with the highest c_j value.

It is difficult to estimate the coverage of this recommendation algorithm, because it is dependent on our limited sample of user traffic data. However, we point out here that over 99% of the photostreams have appeared with at least 3 other streams in the user sessions from our dataset.

Tag-Based Recommender

The second recommender is based on similarity of photo tags belonging to the streams. The algorithm takes as input the last stream seen by the user and recommends those that are the most similar in terms of content.

1. We use a standard information retrieval approach, where streams are documents and tags are words. We use Okapi BM25, as it has been shown to perform well in similar cases [115]. We create a tag vector for each stream.

2. We compute stream-to-stream similarity for each pair of the streams using cosine similarity between the vectors.

3. Finally, we select the streams to recommend with the highest similarity.

We take into consideration only streams that contain at least 10 different tags, what gives us 1.8 million distinct streams. Additionally, we note that over 90% of these streams have a non-zero similarity with at least one other stream. This may serve as an estimate of the upper-bound of the coverage of this recommendation algorithm. The stream-to-stream similarity is computed off-line in order to limit the latency of the recommender system during the user study.

6.4.3. Centering on a Photo Inside a Photostream

Among all photostreams, some of them might contain pictures of various topics. Because only five images are shown to the user (see Fig 6.1b), they play an important role in order to catch the attention of the user. Therefore, we choose which photos to show to the user for each recommended photostream. First, we split the photostream in batches of photos, and then we choose the ones that are the most related to the last photos seen by the user. Note that the images inside a photostream are ordered by the upload time and that the order is not modified in our selection approach.

The photostreams can contain a large number of photos and cover different themes. It has been observed [46] that users tend to load images in batches, and that photos of the same batch tend to share similar characteristics, for example, tags and description. Following this finding, we first split each photostream in batches based on the photo upload date. To this end, we apply the same method as the one that we used to retrieve sessions from the sequences of pageviews (see Section 3.3.1). The split points will occur when the time difference between two consecutive uploads is more than 25 minutes. The first pictures of each batch are candidates to be shown to the user.

There are many ways to chose a batch that is the most related to the last seen images. Unfortunately, our browsing data is too sparse for this purpose. We therefore use the tags of the photos to choose the most appropriate batch. We aggregate tags of all photos belonging to each batch. As input, the recommender system uses the tags of the last $N_p = 5$ photos seen by the user. Finally, the batch that shares the highest number of tags with the set of last seen photos is displayed to the user. We do not use the same approach of the tag-based recommender (cosine similarity among the vectors), since the system has to work in real time in order to perform the user study, and the computation of the similarity among all the possible batches, results to be very expensive.

6.5. Evaluation

We have introduced two recommender systems: a collaborative filtering method using transitions between streams (CF), and a recommender system based on tags (TB). In this section, we compare the performance of these recommender systems focusing on the experience of the user. We test if the recommended photos are related and interesting to users, and we compare the levels of novelty and serendipity of the provided recommendations. First, we present results of a pilot user study. Second, we present the results of the comparison of the two recommender systems in the main user study. The survey of the comparison is shown entirely in the Appendix B.

6.5.1. Pilot User Study

To recommend photostreams we slightly modify the original Flickr photo-focused interface, shown in Figure 6.5a, to place more emphasis on rec-

[0]Sample Flickr pages from the user http://www.flickr.com/photos/bombeador.

Figure 6.5: The two user interfaces tested in the pilot user study. (a) Original Flickr. Hyperlinks of the photostreams which the current photo belongs to are listed (red box), thumbnails are displayed only for the current photostream (green box). (b) Additional rows of thumbnails. Three rows of thumbnails from other photostreams are shown (red box), centered on the current photo (green rectangle).

ommended streams. To this end, we show photo thumbnails from three additional streams, as shown in Figure 6.5b. Also, to gain space for the additional rows of thumbnails, we hide the map of the place where the photo was taken, visible in Figure 6.5a.

The interface was tested in a pilot user study. Each user, was asked to perform two photo-browsing sessions with each of the two user interfaces described above. The sequence of presentation of the interfaces was randomized for every user. Each photo-browsing session lasted a fixed time of 4 minutes. Users were introduced to each interface at the beginning of each session via written instructions on the screen. After completion of both sessions, the user was asked which of the two sessions they liked most. The user study is implemented as a Google Chrome extension, which manipulates the way Flickr pages are displayed and automatically manages all the steps of the study.

In total, we had 33 participants: male (78%) between 26 and 40 years old (84%). Around half of them declared that they used Flickr "a few times a year" (52%), and only 25% use it "a few times a month". The background of the users was mainly formed by students and CS researcher. The great majority of the users preferred additional rows of thumbnails over the original Flickr interface (79% against 9%, 12% no opinion).

6.5.2. Comparative User Study

We conducted the user study to test the following hypotheses:

H1) A recommender system based on transitions among streams could propose related and interesting streams to the user.

H2) Collaborative filtering allows the user to explore more novel content than tag-based recommendation.

In order to evaluate the recommendation algorithms, we integrated them in the interface presented in the previous subsection. Each user was asked to go through two photo-browsing sessions, using each of the two recommendation algorithms in random order. Each of the photo-browsing sessions lasted 6 minutes and started with a grid of 100 images randomly selected among the top-1000 photostreams with the highest number of suggestions in both recommender systems. Users were able to go back to the grid during the experiment. No task was given to the users, apart from a suggestion to

browse the photos freely. Each session began with written instructions on the screen, describing the task and ended with an evaluation form (see Appendix B for more details). Questions were in the form of a statement, and the subjects were able to express their agreement on a 5-point Likert scale (from "strongly disagree" to "strongly agree"). First, we asked users how *related* and *interesting* the recommended photos were. Relatedness expresses how similar the suggested photos are to the displayed one. Interestingness is related to user curiosity and interests. Recommended items are interesting when they catch ones' attention. Second, we asked users about *novelty* and *serendipity*. Novelty is the capability of the recommender system to suggest unfamiliar and non-obvious items [5]. Serendipity is a related concept, since a serendipitous recommendation algorithm proposes items that are novel but also surprisingly interesting. After completion of both sessions, the participants were asked for a final direct comparison.

Results

In total 40 subjects participated in the study, male (66%) between 26 and 40 years old (89%). Around three quarters of them declared that they "never" used Flickr or "a few times a year" (73%), and only 20% used it "a few times a month".

The random null hypothesis is rejected by a χ^2 test ($p < 5 \cdot 10^{-4}$) for the results of each of the questions. For each comparative result we applied the Shapiro-Wilk normality test. Since the normality null hypothesis was rejected for each distribution, we provide the p-value of the Wilcoxon signed-rank test.

The majority of users agreed (answers: "strongly agree" or "agree") that the recommender systems suggested related pictures (61% for CF, 75% for TB). For both recommender systems, the suggested images were found to be interesting (75% for CF, 69% for TB). Moreover, users considered the collaborative filtering recommender to suggest more novel content (51% for CF, 29% for TB, $p < 0.04$). On average, the collaborative filtering recommender was more likely to provide serendipitous encounters (55% for CF, 38% for TB). However, the two algorithms do not show a large statistically significant difference ($p < 0.11$). In the comments many people reported that they found interesting photos or photographers they liked but did not know. Finally, 44% of the participants preferred CF over TB, 39% preferred TB over CF and 17% did not express any opinion.

Additionally, we analyzed logs of the user study and report on them briefly. On average, during the study users of the collaborative filtering system transitioned between photostreams 8.9 times, while those of the tag-based system transitioned 13.3 times. The average number of distinct photostreams seen per session is 11.2 for CF and 11.9 for TB.

Discussion of the Results

Based on the results of the user study, we conclude that both recommender systems provided related and interesting suggestions of photostreams and photos, which gives evidence in support of hypothesis H1. Moreover, the collaborative filtering recommender provided more novel content, and to a lesser extent also more serendipitous content, which confirms hypothesis H2. This result, has also been confirmed in a recent publication by Bellogín *et al.* [10]. However, in our experiments, this did not result in a significant user preference to either of the recommender systems.

From the log analysis we can see that, due to the fact that the tag-based recommendations are more related and less novel, users are more willing to browse the photos by switching between streams, instead of just browsing consecutive photos of the same stream. However, users on average saw the same number of distinct photostreams in the two sessions, meaning that, in the case of the tag-based recommender system, users encounter streams that they have already seen more often.

6.6. Summary and Discussion

In this chapter we worked with photostreams as content units for analyzing user browsing behavior in Flickr. In particular, we presented the results of an analysis of a large sample of Flickr navigation logs to gain insights into how users navigate between photostreams. To analyze frequent stream topic transitions, we created a stream transition graph from over 100 million pageviews. We found interesting browsing patterns in how users navigate between streams and showed that users tend to browse related streams.

Furthermore, we used these findings to design two photostream recommender systems, one based on collaborative filtering (using transitions between photostreams) and one based on content (using photo tags). Both algorithms are part of a two-level photo recommender system that first recommends photostreams, and then particular photos from the chosen photostreams. The recommender systems are computationally inexpensive.

We compared the two recommender systems through a user study involving 40 participants. The majority of users found the recommended photos to be interesting and related. Moreover, the results of the survey confirmed that the collaborative filtering method based on transitions between streams provides more novel recommendations than the tag-based method. The user studies were useful in gaining insights on the functionality that can be provided. The feedback obtained was mostly positive, making the approach very promising.

In summary, we have shown how to recommend items (*i.e.*, photostreams) exploiting the current user's navigational session. The previous users' browsing sessions are used to rank the photostreams that will be recommended to the user. The system we described works also for newcomers, *i.e.*, cold-start, since it learns the interest of the user during the navigation without the need of any user profile. In the next chapter, we will face the cold-start case exploiting the referrer URL and the *BrowseGraph*, we will recommend the next page visited by the users immediately after they have landed to the website.

Chapter 7

News Recommendation Based on the BrowseGraphs

Previously, in this thesis, we observed how the external referrer URL allows us to characterize the type of user navigation (Chapter 4). In Chapter 5, we built the *BrowseGraph* and different *ReferrerGraphs*, in order to study their reliability with centrality-based algorithms. In the previous chapter instead, we discussed and studied how the users browse a website with the aim of discovering and consuming media content (*i.e.*, photos). We implemented a collaborative filtering recommender system, based on the previous users' sessions that could be applied also to the cold-start problems of newcomers.

In this chapter, we focus on how users consume news articles with respect to their browsing sessions. First, we perform a deep analysis on how users consume news articles from a large navigation log of Yahoo News ($0.5B$ entries). Then, we extend the experiments on the *ReferrerGraph*, namely the subgraph induced by the sessions with the same referrer domain. The structural and temporal properties of the graphs show that browsing behavior in news is highly dependent on the referrer URL of the session, in terms of type of content consumed and time of consumption. We build on this observation and propose a news recommender that addresses the *cold-start* problem: given a user landing on a page of the site for the first time, we aim to predict the page she will visit next. The aim is to show how the insights given by the *ReferrerGraphs*, can be used to personalize the content for new users that had never visited the website before. We test this by using different *BrowseGraph*-based and *ReferrerGraph*-based approaches on Yahoo News. The results of this chapter were published in [99].

7.1. Introduction

In recent years the consumption of online news has increased rapidly, in contrast with the decline of traditional newspapers.[1] Between 2009 and 2012, the percentage of users visiting news portals, have raised steadily up to the point to represent the major portion of overall web traffic.[2] comparable to the volume of visits to top domains like Google search.[3] Due to its importance, richness of content, and abundant user participation, the field of online news has become a crowded arena for research in several areas. Some examples are information and multimedia retrieval, ranking, recommendation, and personalization [39, 23, 113]. Despite the vast amount of work in the field, there are two aspects of news consumption that are still largely unexplored. First, modern online news providers have turned into globally connected systems that are able to attract a wider audience than their core of regular users. News articles are very often shared on different external websites and social media platforms, thus providing a growing number of browsing shortcuts to news portals. To mention two examples, modern search engines serve queries relevant to news stories by directly featuring news articles from major providers, and social media is increasingly used as daily tools for journalists and casual news readers,[4] who spread and consume news provided by external parties [23, 78, 104]. Despite such increasing level of integration and mashup, news portals have been studied mostly in isolation. An aspect that has drawn very little attention is the user *browsing behavior*. Although recent literature is rich in studies about browsing patterns in several online platforms [80, 59, 100], little has been done with respect to the news domain. One reason for this is that the browsing sessions in online news outlets have been found to be short, aimed in most cases to quick catch ups on news [59].

In this work, we address these two aspects in combination and exploit them in a task of news recommendation in a *cold-start* scenario. We study the way users browse news content in relation to the *type* of online domain they were browsing *before* landing on the news page, also known as *referrer* domain. Our contribution begins with finding that browsing is a meaningful phenomenon to study also for online news, as the browsing graphs have, in this case, a coherent and well-formed structure. We find that the referrer partly

[1] http://stateofthemedia.org/2012/overview-4/key-findings/

[2] http://www.people-press.org/2012/09/27/section-2-online-and-digital-news-2/

[3] http://www.theguardian.com/news/datablog/2012/jun/22/website-visitor-statistics-nielsen-may-2012-google

[4] http://bit.ly/1bQG1uL

Figure 7.1: An article page from Yahoo News (compacted layout). Right rail boxes and the infinite-scroll section at the bottom allow the user to browse to other articles.

determines the type of news consumed and the time when it is consumed. In the wake of previous studies about the impact of the referrer domains on user browsing behavior [32], we use the browsing graph induced by the referrer of the browsing sessions, to predict the next article a newcomer will visit right after she lands on a page of the site. Using a very large sample of navigation log from Yahoo News ($\sim 500M$ entries), we compare 24 flavors of recommenders for next-article consumption, including popularity, item, and browsing based models. We find that browsing based recommendations achieve the best overall precision@1 among all the methods: up to 48% in conditions of heavy volatility of news articles and of high data sparsity, arising from the large amount of candidate articles to recommend.

We summarize our main contributions as follows:

- We introduce the *BrowseGraph* in the context of news and we define the notion of domain-dependent *BrowseGraph* (*ReferrerGraph*), namely a graph composed by the browsing sessions of users coming from the same referrer domain, *e.g.*, *facebook.com*. (Section 7.2).
- We study the *BrowseGraph* built from a large sample of browsing activity from Yahoo News. We explore it with respect to time and topic, providing insights on relations between the domain of origin, the type of news consumed, and the temporal patterns of consumption (Section 7.3).
- We provide a method to recommend the next article to read in a cold-start scenario, using the information from the *ReferrerGraphs*. Our recommender outperforms a number of item, popularity, and browsing based baselines (Section 7.4).

We are not aware about any other work that uses the *BrowseGraph* for news recommendation. Moreover, we introduce the *ReferrerGraph*, and study in detail to show how the information obtained, helps to improve the accuracy of the recommendation.

7.2. BrowseGraph in the News Domain

To study the activity of news consumption and browsing, we analyze the Yahoo News navigation log, while also considering the browsing activity of users coming from different (*families* of) domains. In this section, we describe the raw data we use and its preprocessing (Section 7.2.1) and how we leverage it to generate the *BrowseGraphs* (Section 7.2.2).

7.2.1. News Website Navigation Log

Each article page in Yahoo News contains a nearly inexhaustible variety of options to perform transitions to other article pages. As shown in Figure 7.1, the typical news page contains a right rail with several boxes of recommended news (*e.g.*, recent articles), and an infinite-scroll list of personalized news at the bottom. To capture the user's browsing activity we consider the dataset of log data, described in Chapter 3 (Section 3.3.2).

7.2.2. Domain-Dependent BrowseGraph

We go beyond the study of the general browsing patterns, by comparing the browsing behavior of users who land on the news website from *different domains*. We aim to verify empirically the idea that users coming from different types of external web services are interested (or exposed) to different types of content, and therefore behave differently.

To decompose the overall *BrowseGraph* G into subgraphs G_d, that account for the sessions originated from a specific domain d as explained in Section 3.3.2, we use only the sessions whose first referrer URL matches the domain d. For instance, a user who accesses a news page from a tweet (*i.e.*, Twitter message), will start a new session that will be part of the Twitter *ReferrerGraph*. We refer to the subgraph G_d as the *ReferrerGraph* for the domain d. In particular, we consider 9 source domains. Three search engines: *Bing*, *Google*, and *Yahoo*; three social networks: *Facebook*, *Reddit*, and *Twitter*; and the *homepage* of the news portal, a special case of referrer URL that represents a significant entry point for Yahoo News users. In addition, we also consider two aggregated *ReferrerGraphs* created by the union of the graphs of *search engines* and *social networks*, respectively (we call them *Search* and *Social*), as some of the characteristics of the *ReferrerGraphs* of domains belonging to the same family are quite similar (see Section 7.3).

Last, since the consumption of news items is strictly dependent on time, we define a *temporal BrowseGraph* G^t as the *BrowseGraph* originated by the browsing sessions occurring in one hour t. We partition the *BrowseGraphs* on hourly intervals, ending up with $1,440$ temporal graphs for each domain, for a total of $12,960$ graphs.

7.3. Analysis

In this section, we report the structural properties of the full BrowseGraph and of the *ReferrerGraphs* induced by the different domains of origin (Section 7.3.1), and a study of their evolution in time (Section 7.3.2).

7.3.1. Domain-Dependent News Consumption

We find the distribution of the number of hops per session, to be broadly distributed (not shown) but with very low average values (Table 7.1), in agreement with previous work that found the user interaction, with news portals, being short and time-constrained [59]. Despite that, the Browse-

Full	Homepage	Google	Yahoo
1.94	3.11	1.81	1.97

Bing	Facebook	Twitter	Reddit
1.79	1.34	1.24	1.12

Table 7.1: Average number of hops during browsing sessions with different referrer domains.

Graph built from the full set of user sessions is connected, with a greatest weakly connected component that spans up to 95% of all the pages, and whose nodes are ∼5 hops away on average (statistics are summarized in Table 7.2). This means that, although the individual browsing interactions with the news portal are short, the collective browsing behavior weaves an implicit network of associations between articles whose points are on average 5 hops away. The connectivity of the BrowseGraph appears to be scale-invariant, as very similar connectivity values are found for the *RefererGraphs*, the most disconnected one being Twitter, with 87% of nodes in its giant component. The average in-degree can be considerably high, due to the large number of possible connections that an article page has with others (as illustrated in Section 7.2), and it is by far the largest in the *homepage* graph, which dominates in terms of traffic volume. Although the degree distributions vary considerably (Figure 7.2), the edge weight distributions are closer to each other, and the vast majority of edges have very low weights.

A natural question is whether the graphs are different just in terms of structural properties, or also with respect to the *type* of their nodes. To measure the overlap between graphs, we compute the Jaccard similarity between the set of their nodes (Figure 7.3a). As one might expect, similarity is lower between the two major families: search and social. Surprisingly though, there are conspicuous differences also within each group. For instance, Twitter and Reddit have only ∼20% of the overall amount of their nodes being covered by both. This means that the users coming from Twitter are visiting only a small portion of the news articles visited by users coming from Reddit. In other words, the users' interest is strongly dependent by the type of website they are coming from. We spot also significant differences in the type of news content consumed in the different networks. To measure that,

Graph	#Nodes	#Edges	Density	%GCC	$\langle k_{in} \rangle$	$\langle d \rangle$
full	745,720	10,017,826	$1.8 \cdot 10^{-5}$	0.95	2551	5.14
homepage	257,465	3,516,661	$5.3 \cdot 10^{-5}$	0.99	1830	4.15
google	163,411	928,364	$3.5 \cdot 10^{-5}$	0.93	400	3.98
yahoo	116,403	490,239	$3.6 \cdot 10^{-5}$	0.95	229	2.91
bing	70,665	308,824	$6.2 \cdot 10^{-5}$	0.91	224	3.34
facebook	24,058	84,837	$1.6 \cdot 10^{-4}$	0.95	141	3.31
twitter	5,065	8,922	$3.5 \cdot 10^{-4}$	0.87	39	3.17
reddit	2,840	5,851	$7.3 \cdot 10^{-4}$	0.95	81	3.67

Table 7.2: Structural statistics of the *ReferrerGraphs* ($\langle d \rangle$ indicates the average shortest path length and GCC indicates the Giant Connected Component).

we count the frequency of articles belonging to each of the news topics (see Section 7.2.1), and we rank topics by their frequency in each network (Table 7.3). The rankings show substantial differences, with celebrity-related news being the main interest for users coming from search engines, while blogs, sports, and entertainment are the most popular topics in Facebook, Twitter and Reddit respectively.

The differences in terms of graph structure, their size and type of nodes, impact directly the type of articles that are consumed the most, or that are perceived as most interesting by the users. To gauge that, we consider two metrics of news importance, namely the *pageview count* (*i.e.*, *view rank*) and the *PageRank centrality*, computed on the weighted graphs. We apply each metric separately to the *ReferrerGraphs*, and we compute pairwise the Kendall τ similarity between the ranks. Figure 7.3b displays the values for PageRank. To discount for the different dimensionality, the Kendall τ is measured only on the elements contained in the intersection of the two sets. To account for the noise that can be potentially introduced by the permutations on the latest positions on the rank (*i.e.*, articles with very similar scores in the long-tail of the popularity curve), we repeat the same measure on the top 100 and 1000 articles, obtaining very similar results. Rankings tend to be more similar within domain families ($\tau \in [0.50, 0.68]$ for search and $\tau \in [0.20, 0.27]$ for social) than across families ($\tau \in [0.14, 0.4]$).

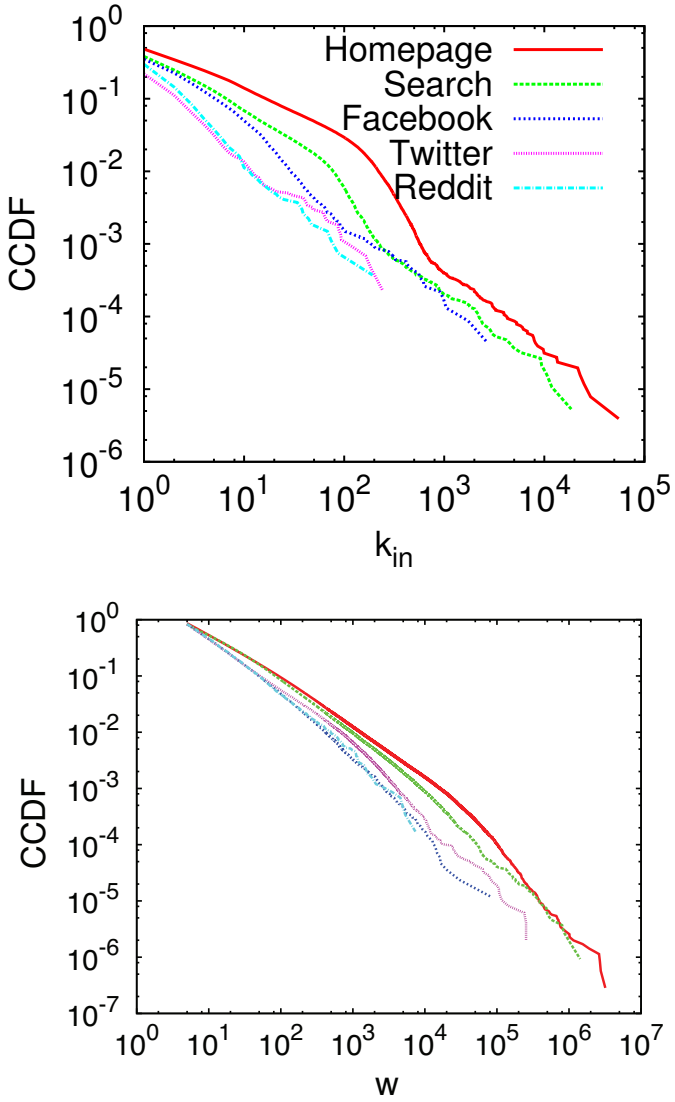
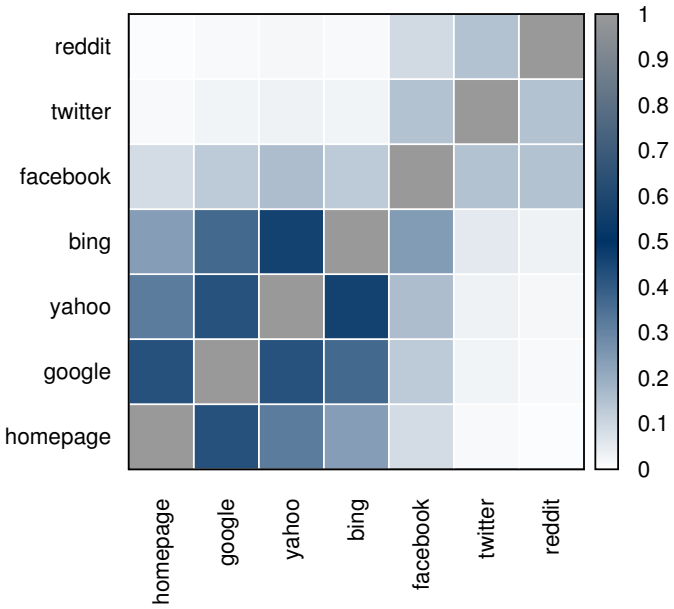
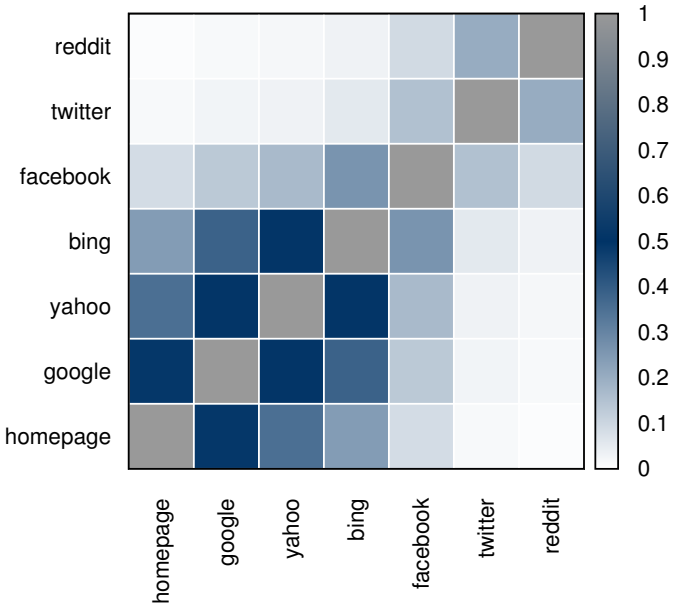


Figure 7.2: Distributions of indegree (k_{in}) and edge weight (w) in some *ReferrerGraphs*. Search graphs are collapsed in one curve due to their similar distributions.



(a) *Jaccard similarity of node sets*



(b) *Kendall τ between news pageranks*

Figure 7.3: Node overlap between graphs and article ranking comparison.

7.3.2. News Consumption in Time

Time and Domain

Time plays a central role in news content consumption as news articles tend, by their nature, to become rapidly stale. In Figure 7.4 (left) we plot the distribution of the relative volume of views that articles receive in time (hours). In Figure 7.4 (right) we show the same measure but on a normalized time axis, that starts from the article publishing time ($t = 0$) up to the last visit received ($t = 1$). We refer to this normalized timeline as the article *lifespan*. Consistently with previous work, we find that 80% of visits are received within the first 30 hours after the article publication and before the first 20% of the overall article lifespan.

One question, however, is whether the temporal aspect of news consumption depends on the type of network the user is coming from. To investigate this aspect, we repeat the previous measures separately on the three *ReferrerGraphs homepage*, *search*, and *social*. For each of them, we measure the distribution of the total volume of visits at each point of the article normalized lifespan (Figure 7.5). A phenomenon of rapid decay emerges in all three cases, however the curves exhibit major differences in skew. While news consumption through the homepage tends to happen in earlier stages of the lifespan, accesses through social networks and search engines are shifted towards later stages. In particular, for the *social* domain we observe evident peaks of accesses during late stages of the article life. Even if these peaks account for rather a small percentage of the whole traffic (up to 2%), they still represent a non-negligible number of accesses. Examples of news, that largely contribute to the visit volume inside the peaks, belong to the "trivia" type of stories that are most commonly seen on social networks.[5]

Time and Topic

The referrer domain is just one variable that might impact the temporal patterns of content consumption for news, and the type (or topic) of the article can also have a role on that. In Figure 7.6 we show the aggregated volume of views in time for news articles, belonging to six different categories. The relative positions of the accesses from homepage, social sites, and search engines, change depending on the topic. The baseline consumption behavior is given by the general type of news, for which the view volume from the homepage is consistently higher than the volume from search engines, which

[5]See, for instance: http://abcn.ws/1fPc0zu, and http://abcn.ws/1iX4nHD

Full	Homepage	Search	Facebook	Twitter	Reddit
Celeb.	Video	Celeb.	Entertain.	Sports	Blogs
Finance	Celeb.	Finance	Celeb.	Finance	Politics
Video	Finance	Video	Video	Video	Sports
Sports	Sports	Sports	Finance	Entertain.	Technology
Politics	Politics	Movie	Blogs	Lifestyle	Finance
Movie	Movie	Politics	Sports	Movie	Movie
Lifestyle	Lifestyle	Blogs	Photos	Photos	Video
Blogs	Blogs	Music	Lifestyle	Celeb.	Lifestyle
Music	Music	Entertain.	Movie	Music	Celeb.
Entertain.	Entertain.	Lifestyle	Politics	Politics	Health

Table 7.3: Top categories for different subgraphs.

is in turn higher than the one from social networks. For blogs instead, the view volume is more similar across the three macro-networks and the curves intersect more often, with two clear phases. First, the number of visits from social networks goes above search for a short time, likely explained by the fact that blog posts in important news sites are usually written by bloggers, who are heavily involved in the activity of online social networking. Last, the accesses from homepage and search become comparable in volume. Similar observations hold for other categories: sports, movies, and celebrities get a higher volume of accesses in later stages of the article life. In the case of celebrities in particular, we observe that accesses from social exceed even the ones from the homepage, after a certain point. This may happen when news about specific events (*e.g.*, academy awards) cause an outburst in the social media discourse. Last, an interesting case emerges from visual news such as photo-galleries related to news events. In this case, the accesses from social and search are comparable in the early life of the news, while the volume from search and homepage are comparable in the later stages. This delineates a scenario in which images lend themselves to spread easily in social media.

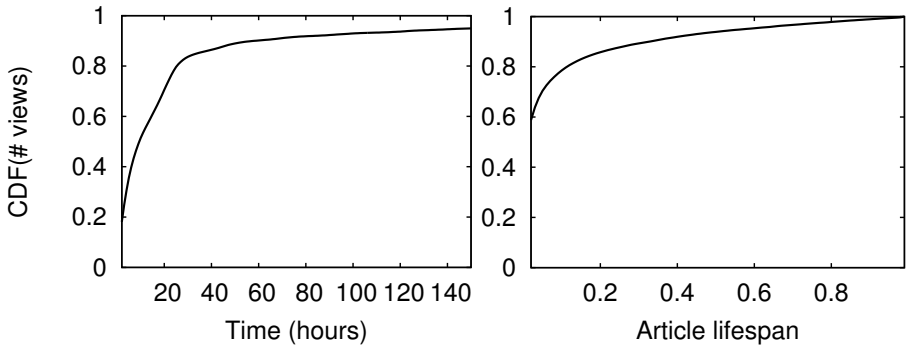


Figure 7.4: Cumulative number of page views in time.

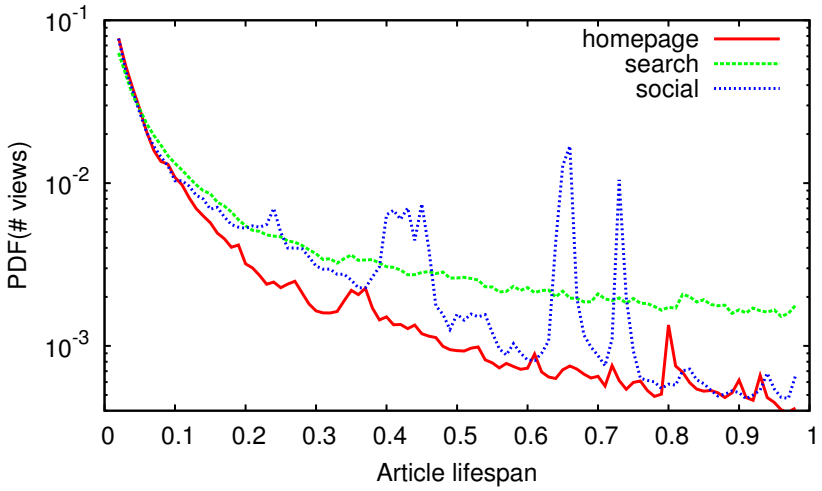


Figure 7.5: PDF of the number of views received in each of three *ReferrerGraphs* over the normalized lifespan of the news, from the publication ($x = 0$) to the last visit ($x = 1$).

Rank Variation

Besides studying the attention received in time by the overall set of articles, it is interesting to check how the attention received an article changes in relation to others. In other words, if we rank articles by viewcount, we can explore how the rank changes in time and across *ReferrerGraphs*. To quantify that, we consider *Homepage*, *Search*, and *Social ReferrerGraphs* separately, and for each of them, we compute an hourly view rank R_t for

all the articles they contain. Then, for each set of articles published in the hourly time slot t_i, we compute the Kendall τ between their view rank at t_i, and the view ranks in subsequent hours. More formally:

$$\tau(R_{ti},\ R_{ti+j}), \qquad \forall j \geq 1$$

Then, we shift each measurement back in time by i hours, so that all sets of articles start from time 0, and we average all the measurements, with resulting curves in Figure 7.7. The lower the value of τ, the farther the ranking at time t is from the ranking at the original publication time. In all the cases, we observe the values decrease rapidly in the first 5 hours and a steady state occurs within the first 24 hours. This finding is consistent with the volume of views dropping of several orders of magnitude in few hours. The τ value after 2 days is, on average, not higher than 0.55, meaning that the final ranking changes considerably from the initial one. Although the trend of the three curves is analogous, they have different offsets. Articles accessed via search change their relative position less and the ranking stabilize slightly quicker, while on the other extreme, accesses from social networks impact more the rank based on the number of views.

7.4. Cold-start Prediction of Next View

Item recommendation is a crucial task in news sites, as they have to deal with a rapidly changing pool of thousands of fresh articles and millions of users, each one with a specific range of interests. In such a scenario, profiling users with their explicit (*e.g.*, comments, article saved, printed, shared) and implicit (*e.g.*, views, time spent) activities on-site is an effective way to recommend new content that matches the user interest. However, personalization is not possible in cases of *cold-start*, when a user who is a newcomer or is not logged-in lands on the site. In this context, the information of the *BrowseGraph* can help, as the activity of previous users provide a collective trace of previous browsing patterns, that can be recommended also to the new user. In particular, we show that the *ReferrerGraphs* are particularly effective to this end. Next, we formally define the recommendation problem (Section 7.4.1), describe a number of methods to address it (Section 7.4.2), compare them on a large scale dataset (Section 7.4.3), and finally discuss the results.

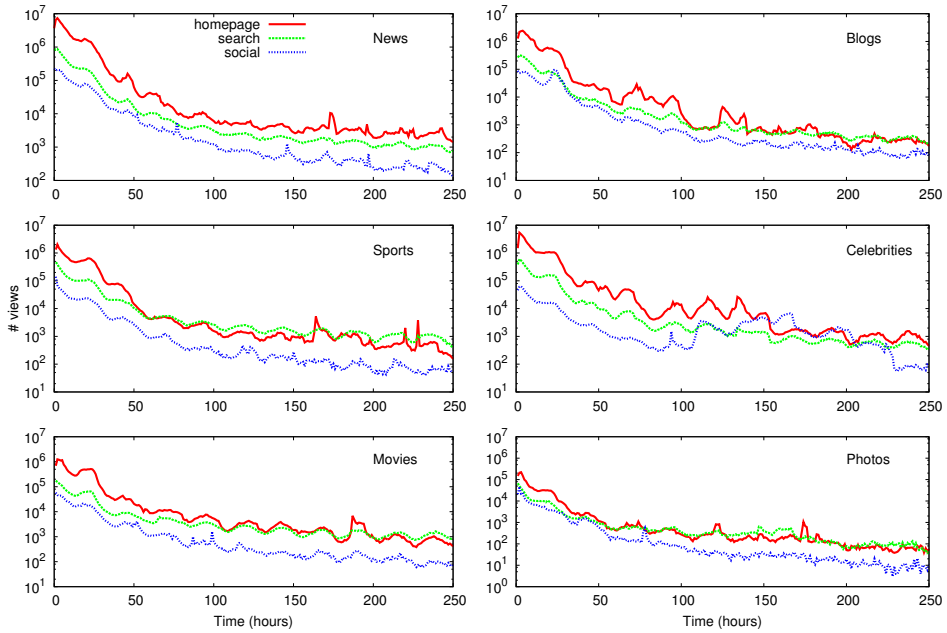


Figure 7.6: Number of views in each *ReferrerGraph* in time, breakdown by news topic.

7.4.1. Problem Definition

The prediction problem we address is defined as follows. A newcomer user $u \in U$ is given, who begins a new session at time t on page $p_{start} \in P$, with referrer domain $d \in D$. The task consists in predicting a page p_{next} that u will visit right after p_{start}. Note that, we restrict the problem to users whose sessions will include at least two pageviews, *i.e.*, an additional pageview after p_{start}. We consider, for simplicity, a time line quantized in discrete 1-hour slots, and we assume to know the information about the browsing sessions generated by other users in the previous time slot $t-1$. To be able to draw a comparison also with recommendation methods based on textual content or item popularity, we consider an additional set of meta-data for every page $p \in P$. Specifically:

- v_p^{t-1}: cumulative number of pageviews at time $t-1$;
- cat_p: the page's topical category;
- h_p: the page textual headline;
- b_p: the textual body of the page.

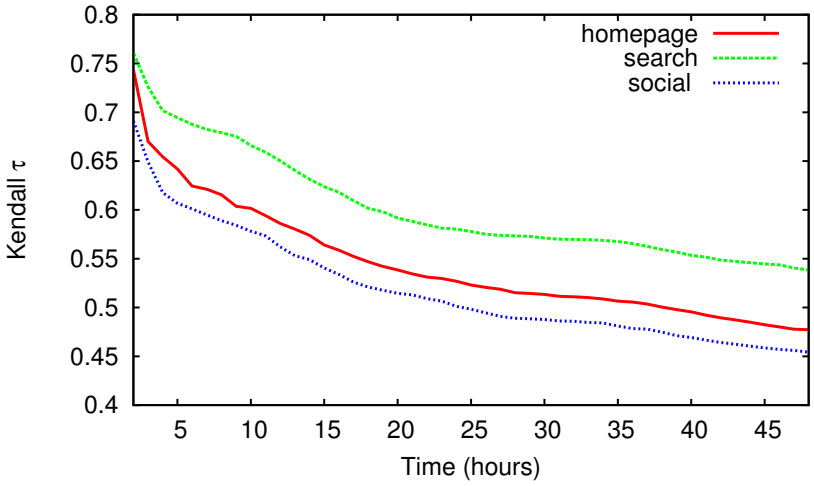


Figure 7.7: Kendall τ calculated between the view rank at time t and the view rank at time 0.

7.4.2. Prediction Methods

All the prediction algorithms we consider are determined by the combination of three components we describe next.

Selection of Candidate Pages

Full neighbors set (full). After the initial visit of p_{start}, the target user could transition, in principle, to any other page in P. However, we measure that in the 95% of the cases, p_{next} is included among the set $\Gamma_G^{t-1}(p_{start})$, namely the out-neighbors of p_{start} in the *BrowseGraph*, created from the browsing sessions occurring during the time slot $t-1$. Formally:

$$\Gamma_G^{t-1}(p_{start}) = \{p_i | (p_{start}, p_i) \in E(G^{t-1})\}$$

This happens because, even though the cardinality of $\Gamma_G^{t-1}(p_{start})$ can be very big (recall the degree distribution in Figure 7.2), most of the browsing links between p_{start} and its neighbors are created shortly after the news is published. For this reason, an effective strategy would be to consider only the set $\Gamma_G^{t-1}(p_{start})$ as output range for the prediction. We call this selection strategy **"full"**, after the full *BrowseGraph* we use to perform the selection.

Referrer neighbors set (ref). In Section 7.3 we observed that the type of referrer URL determines, to a certain extent, the type of news consumed.

One might argue that adapting the candidate page selection based on the user domain of origin d, could potentially improve the prediction accuracy. We could therefore restrict the output range, to the neighbors of p_{start} in the domain-dependent graph G_d^{t-1}. Using G_d^{t-1} instead of the full graph G^{t-1}, implies a drop in the chance of finding p_{next} in the set $\Gamma_G^{t-1}(p_{start})$, from 95% to a minimum of 48% for the Yahoo graph, and a maximum of 72% for Homepage. However, based on our previous analysis, the subset of remaining pages could have a higher likelihood of being good candidates for prediction. Similarly to the previous case, we name this selection strategy **"ref"**.

Mixed neighbors set (mix). A natural extension is to combine the **ref** and **full** approaches in cascade. By definition, G_d^{t-1} has a subset of the nodes in G^t. Therefore, it may happen that the node p_{start} is not present in the subgraph or does not have any out-neighbor. So, we adopt the following strategy: if $p_{start} \in N(G_d^{t-1}) \wedge k_{out}(p_{start}) > 0$, then use the **ref** strategy, otherwise rollback to **full**. In the following, we refer to this strategy as **"mix"**.

In the remainder of this chapter, we call C the set of candidate nodes, disregarding the strategy used to obtain it.

Topical Filtering

As we report in Table 7.4, the probability of transitioning from a page with a topical category (cat_p) to another page with the same category, computed over all the sessions, varies depending on the domain of origin for that session. In the cases of *Twitter* and *Facebook*, there is a slight tendency to stick to the same topic, whereas for the other domains two consecutive pages in the session tend to belong to different categories. We leverage this information to enrich the initial candidate selection strategy, keep only those articles in C that belong to the same topic as p_{start} for *Twitter* and *Facebook*, or to a different topic for the other domains.

Prediction of Next Page

All the methods, use the *BrowseGraph* information to select an initial set of candidate pages C, according to one of the strategies defined above. After that, a criterion for the selection of the predicted next page among the ones in the set is needed. Next, we describe four algorithms, with their shortnames in parenthesis.

Total	Facebook	Twitter	Reddit	Google	Bing	Homepage
0.34	0.59	0.64	0.48	0.44	0.44	0.33

Table 7.4: Probability that a user navigates between pageviews of the same category.

Random (rand). A simple baseline that selects at random a node in C.

Content-based (cb). A standard approach to recommend items at cold-start is to select the most similar article to the one the user is currently consuming, according to text-based metrics. When the body of the article is available (35% of articles in our dataset) the similarity is computed between the bodies, otherwise their headline (always available) is used. We compute the cosine similarity of the vector representation, weighted with TD-IDF of p_{start} with the ones of every $p_i \in C$. Text is preprocessed with stopword removal and stemming [112].

Most Popular (pop). Another typical cold-start recommendation approach is to select the most popular item. We recommend the node in C with the highest view count, considering the views until time $t-1$.

Edge-based (edge). Consider the weight on the edges that encode the likelihood of a transition between nodes, according to the browsing traces recorded at time $t-1$. Hence, we predict p_{next} to be the node in $\Gamma_G^{t-1}(p_{start})$ with highest weight on the incoming edge from p_{start}. Depending on the initial candidate selection strategy (Section 7.4.2), the edges considered (and their weights) will be either the ones in the *BrowseGraph* (for the **full** selection) or the ones in the *ReferrerGraph* (for the **ref** selection).

7.4.3. Experimental Results

We apply our prediction strategies to the sessions of $1,438$ hourly time slots, for an average of $350K$ users per time slot. We evaluate the goodness of the prediction by measuring its overall **Precision@1**: a true positive occurs when the predicted page is equal to p_{next}, a false positive when that condition does not hold. Additionally, since all the methods we presented lend themselves to produce a ranking of pages (based on popularity, similarity, *etc.*), we also measure their Mean Reciprocal Ranking for the top 3 news articles (**MRR@3**). As noticed earlier, there is always a chance that the correct article cannot be possibly predicted because p_{next} might be

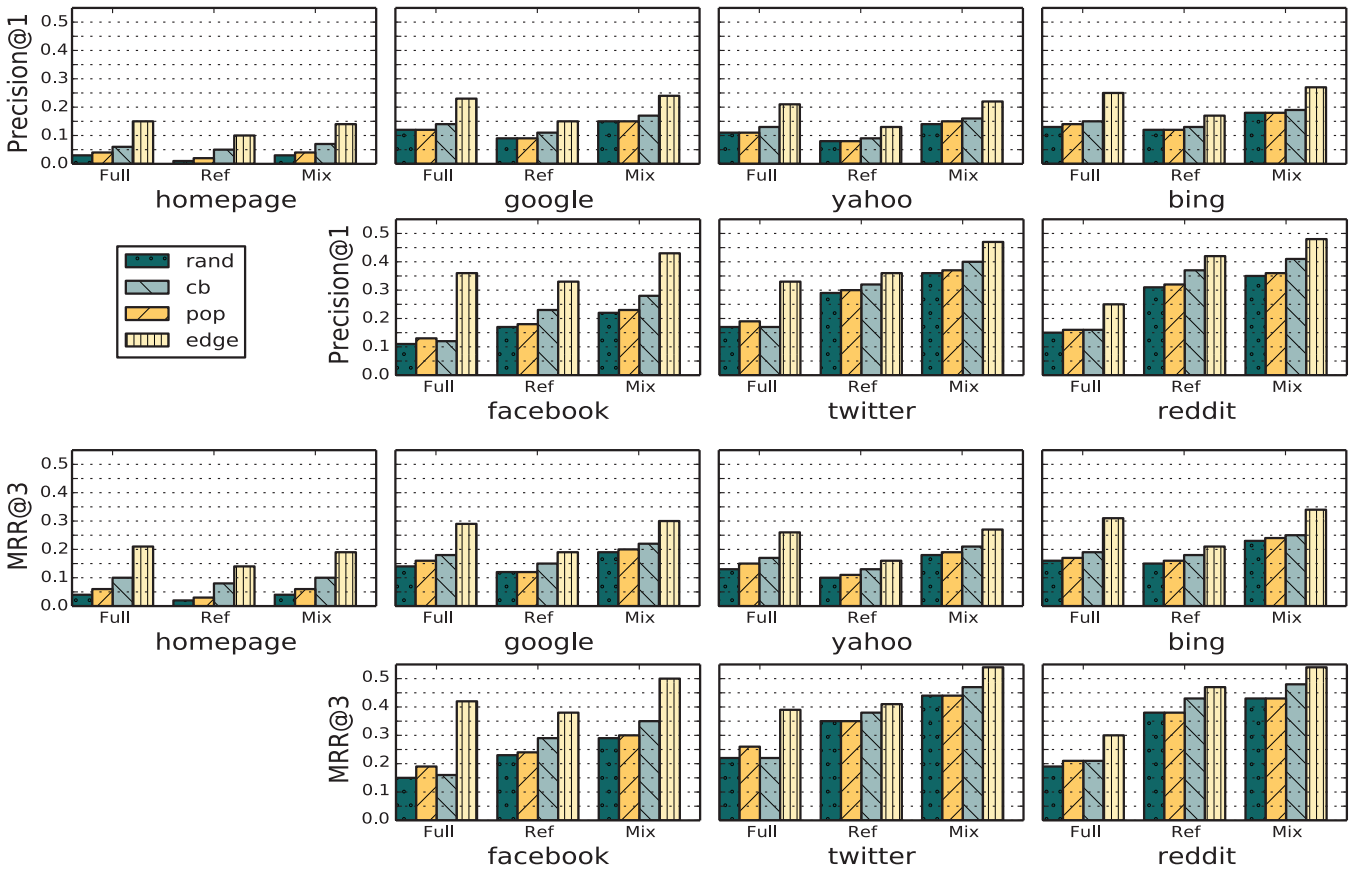


Figure 7.8: Prediction accuracy for the 9 recommendation strategies, computed for the sessions in each *Referrer-Graph* separately.

not included in the set of candidates $\Gamma_G^{t-1}(p_{start})$ (for example because the article is published at time t and does not exists yet at time $t-1$). We adopt a conservative approach and we count also these cases as false positives. Figure 7.8 summarizes the prediction results. To have a more detailed picture of the cases in which the different approaches work best, we report separate evaluation results for the sessions with different referrer domains. Twelve bars for each group represent the precision and MRR scores, for the combinations of the three selection strategies (**full**, **ref**, **mix**) with the next-node selection methods (**random**, **cb**, **pop**, **edge**). The maximum precision achieved for the different domains, partially depends on the dimensionality of the session volume for that domain. This is mainly because the smaller the $\Gamma_G^t(p_{start})$ set, the higher the probability of getting a correct prediction just by chance. The most interesting experimental findings, lie instead in the offsets between different methods' results within each referrer domain.

First, the random baseline achieves always the worst performance, followed by the content, popularity and edge strategies, in order. About the low performance of **cb**, our hypothesis is that the selection of next article is not driven by patterns of content similarity. In other words, after having read an article on a topic the user is likely not motivated to keep reading about the same (or similar) topic right after. The **pop** approach works only slightly better, as it relies on aggregate information about the amount of page visits, but disregarding where such visits came from. The best method by far is **edge**, meaning that previous transitions from p_{start} and p_{next} constitutes the stronger signal for the prediction of future transitions. For the social referrer domains it is able to reach up to 48% and 54% in P@1 and MRR@3. For the search domains instead, it reaches up to 27% and 34%.

Regarding the node selection strategies, **ref** outperforms **full** in all the three social domains (except for the **ref-edge** combination in Facebook). The fact that a more specific type of recommendation works better, suggests that people coming from social networks tend to retrace the same browsing paths that other people from the same referrer domain have already explored, with limited serendipitous discovery. The opposite occurs for the search domains, where **full** beats **ref**. This may happen because query-driven systems provide a wider range of entry points to the news site than the links posted on social networks, thus making the prediction task harder. The same happens for the homepage, where the variability of content displayed is very wide and dynamic. However, for both families, **mix** is the most effective strategy that is able to significantly boost even more the precision for social networks, and to fill the performance gap with **full** for the search domain.

Homepage, which is the domain originating the highest number of sessions is the only one in which **full** has top precision. In this case, the behavior of users is so varied, that restricting the options to a subgraph turns out to be detrimental for the prediction quality.

Last, when the topical filtering is applied (Section 7.4.2), the precision experiences a drop in performance loosing from 10.6% up to 69.5% (not shown in plots). This happens because discarding too many nodes introduces the high risk of ruling out very good candidates (*e.g.*, a node connected to p_{start} with a high-weight edge). In our case, as shown in Table 7.4, the probability of transitioning to the same topic (or to a different one) is not far from 0.5 in all cases, therefore the topical information is not discriminative enough to filter out nodes without losing the most likely next pages.

The experiments highlight how the referrer URL of a browsing session can help to understand the user behavior, predicting the navigation pattern and improving the next-hop recommendation in news browsing. A recommender that uses the weights of the *BrowseGraph* edges, appears to be an effective way to anticipate user needs and keep them longer on site, especially for people coming from social media. This is particularly important, as it has been shown [23, 78, 104] that social media platforms are playing an increasingly important role on news propagation[6], and they are meant to become even more critical connections with the news world in the near future.

7.5. Summary and Discussion

We presented an analysis of the browsing traces extracted from a very large navigation log from Yahoo News, introducing the definition of a special case of the *BrowseGraph* model, namely the *ReferrerGraph*, that consists of a subgraph built from the browsing sessions with homogeneous referrer URL. We find that the browsing graphs of news sites are well-connected despite the tendency to rapid staleness of content and to the typically short user sessions. *ReferrerGraphs* built considering 9 major domains, appear to be quite non-overlapping, to cover articles of different topics, and to lead to the emergence of different sets of most popular articles. Traffic traces coming from different families of referrer domains have different time consumption patterns: for example, the sessions originating from search engines and social networks, tend to consume content slightly after the visits coming from the news site homepage, and with some bursty consumption peaks for social

[6] http://www.journalism.org/2013/10/24/the-role-of-news-on-facebook/

networks caused by occasional spread of viral stories. Last, we build on our analytical findings by showing that the *ReferrerGraph* can be used for effective article recommendation in a cold-start scenario. As our goal is limited to the prediction of the *next page* visited, more general content-based techniques for cold-start [2, 103] are not directly comparable with our approach, although a more extensive comparison would be valuable to gain a broader view on the problem. At any rate, the findings highlighted in this chapter should lead to a greater consideration of the referrer domain with particular focus on cold-start problems.

In the remainder of this thesis, we will compare ranking approaches based on user explicit and implicit information. We will see how the referrer URL gives an important contribution depending on the type of rankings we want to achieve.

Part III

Implicit and Explicit Information

Chapter 8

Explicit Information in Flickr

In this chapter we analyze the most common user explicit preference in Flickr, called "favorite", that in general, represents a positive opinion from a user regarding a specific photo or video. Liking or marking an item as favorite is one of the most pervasive actions in social media. This particular action plays an important role in platforms where a lot of content is shared. In order to gain insights on the liking behavior in social media, and to inform strategies for recommending items that user may like, we take a large sample of users in Flickr, and analyze the logs of their favorite actions, considering factors such as time period and social connection. Finally, we also perform recommender experiments using this explicit signal.

In Chapter 9 we will compare algorithms and strategies based on implicit information, with ranking approaches based on explicit feedback that we discuss in this chapter.

The results of this chapter were published in [69].

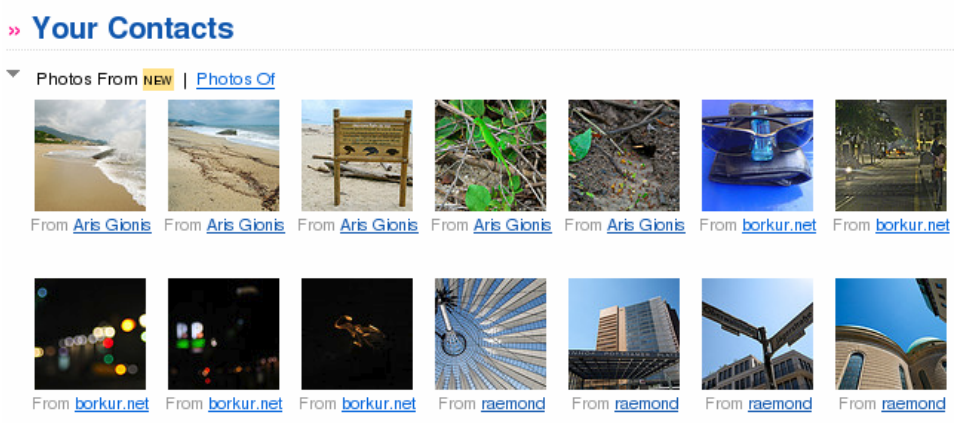


Figure 8.1: A panel in the Flickr interface that presents a set of 14 most recent photos from the contacts of the user.

8.1. Introduction

Sharing and marking objects as favorites are fairly new phenomena in social media, and many questions remain open on the behavior of users in relation to liking or marking an object as a favorite. Questions include *what* they favorite, *when* they favorite, and *how* they are related to the owner[1] of such object (*e.g.*, in Flickr users can be "contacts", "friends", or "family").

The problem of understanding the dynamics of such actions is of extreme importance given the pervasiveness of sharing, and like/favorite actions, in many social media platforms. It is important because when users express such preferences explicitly, they are implicitly contributing to the building of more accurate user models of themselves. Such models have applications in a wide range of areas: they can be used to recommend content, to improve user experience in terms of interaction design, for advertising purposes, or for recommending other users. In addition, one of the basic functionalities of social media platforms is providing easy access to content added by friends and other types of connections. It is common in these platforms (*e.g.*, Facebook, Twitter), to rank items based on various features such as recency. However, due to the increasing amount of shared content, and the size of personal networks, a simple recency based ranking is insufficient. Gaining insights into the favorite actions can contribute in designing novel ranking and recommendation algorithms, and developing new functionalities around

[1] We use the term "owner" to refer to the user who uploads the photo.

surfacing content users may like or favorite.

In this chapter we present an analysis of favorite behavior on a large Flickr dataset. We analyze over 110 million favorite actions, focusing most of our study on a set of 24,000 users.[2] In particular, we examine temporal factors, user profiles derived from tags, and photo and photo-owner features, as well as the relationship between favorite actions and different link types between the users performing the actions and the owners of the photos. Finally, we perform experiments using several features to gain insights into their suitability, and to build algorithms for recommending photos to "favorite."

We examine the following: *a*) whether users tend to favorite photos of people connected to them, more than of people who are not connected; *b*) whether users tend to favorite recent photos more than non-recent photos; and *c*) whether the favorite activity happens in bursts. In addition, we evaluate several features for predicting favorite actions.

8.2. Related Work

In this section we discuss related work about Flickr and, in particular, about the "favorite" action that we analyze in depth in this chapter.

Valafar *et al.* [105] performed a study of favorites in Flickr and found that 10% of users are responsible for $80-90\%$ of all favorites, and that the favorite action exhibits 50% overlap and 15% reciprocity between users. These statistics are confirmed by many other studies (*e.g.*, [27, 64, 85]). Cha *et al.* [27] investigated how an image spreads through the social network, highlighting how propagation varies considerably with the duration of exposure to new photos. In some cases, it takes a long time for photos to propagate from one user to another (*i.e.*, [26, 28]) as there is an initial phase of exponential growth in the number of users that favorite a photo, followed by a phase of slow and linear growth over the years.

Lee *et al.* [60] studied reciprocity in Twitter, and also in Flickr around favorites, by dividing users into three groups: those that only browse, those that also upload photos but do not participate in social activities, and those that participate in social activities. Van Zwol *et al.* [110] presented a multimodal machine learning based approach that combines social, visual, and textual signals to predict favorite photos, while Lu and Li [75] exploited the

[2] All analysis was aggregate, anonymous, and only on public photos.

photos previously marked as favorites by friends, in order to build a personalized search model to assist users in getting access to photos of interest.

Wonyong *et al.* [40] recommended tags for newly uploaded images, taking advantage of the tags assigned to favorite images of the user who uploaded the image, and combining tags with visual similarity. A similar work presented by Chen *et al.* [119], used favorite photos in order to extract representative tags, under the assumption that favorite images are better annotated.

Gursel and Sen [47] proposed an online photo recommendation system based on metadata and comments, assuming these two sources are highly related to the user's interests. De Choudhury *et al.* [38] developed a recommendation framework to connect image content with communities in online social media. They used visual features, user generated tags, and social interaction (*i.e.*, comment actions) to recommend the most suitable group for a given image.

A significant number of studies have been published using Flickr data, so we focused only on citing those that specifically deal with favorites and that are more relevant to this chapter. We are not aware, however, of any large-scale favorite action analysis such as the one we present here.

8.3. Flickr Dataset

Our dataset consists of a snapshot of Flickr until May 2008, which includes the explicit social network at the time, and all interactions on public Flickr photos: over 110 million favorite actions made by over one million users.

Most users favorited photos only a few times (expected long tail of the distribution), more than $140K$ users favorited at least 100 photos, and the most active users favorited almost 100 thousand photos (head of the distribution). Since long tail and head users are not representative for most of the favorite actions, we discarded them from most of the experiments. As we show later, the origin and recency of photos are very important factors. Therefore, with some exceptions, for the rest of the chapter we consider only favorite actions made on photos uploaded to the system, by the user's *connections* within 10 hours of the favorite action recorded. In order to limit the impact of the extreme cases, we constrain the set of users, for which we run the experiments, to users that have done more than 100 and less than $2,500$ favorites. We refer to this set as the "*sample*". We end up with $24,000$ users that chose 8.6 million favorites among 1.2 billion photos.

Links Type	No. of Favorites	Avg Favorites per Link
contacts	29,642,943	0.90
friends	28,125,595	0.79
family	4,577,669	0.63
any link type	59,206,180	0.83
all favorites	112,177,317	10^{-5} (estimated)

Table 8.1: Social links statistics. *All favorites* includes favorites from users that are not linked by any of the relationship types to the user performing the favorite action.

8.4. Data Analysis

In Flickr, each photo has an *owner*, and users can be linked by more than one relationship type (*contacts*, *friends*, *family*, or any combination of those three). In the rest of the chapter we will use the word *connection* to refer to any of the relationship types, but when we use the word *contact* we refer only to the contact relationship.

8.4.1. Photo Origin

We calculated the number of favorites with respect to link types, see Table 8.1. The largest number of favorite photos come from *contacts*, both in terms of absolute number and average number of favorites per link, while *family* links have the fewest number of favorites. At the same time, nearly half of all favorites come from linked users: users tend to favorite photos of users that they are linked to, especially of their *contacts*.

8.4.2. Recency

Recency of a photo has been found to be important in image retrieval (*e.g.*, see van Zwol [107]). An analysis of our entire dataset of 110 million favorite actions shows that 20% of favorites happen within 10 hours, 30% within 24 hours, and 50% within a week from upload time.

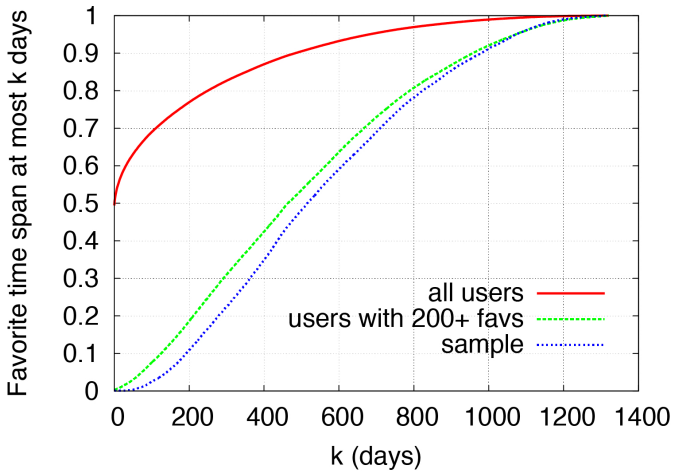


Figure 8.2: Cumulative distribution of the ratio of time span of favorite actions.

8.4.3. Time Span of Favorite Actions

With *time span* we denote the number of days between the first and last time a user favorites photos. We examined the following sets of users:

i. **All users**: the entire set of users in the initial dataset (over one million users).

ii. **Users with over** 200 **favorites**: 80% of all favorite actions are performed by the users in this set, that is obtained by filtering the one million users by selecting only those that have more than 200 favorites.

iii. **Sample**: the set of $24K$ users described in Section 8.3, where we considered only favorite actions performed within 10 hours of uploading of a favorited photo.

Figure 8.2 shows the cumulative distribution of users who performed favorite actions in a time span of k days. The distribution for *all users* is strongly biased by the long tail of users with a small number of favorites. In group (ii), the distribution resembles a normal distribution with high variance, where 80% of users have a time span of over 200 days. In group (iii) the ratio is even higher, around 90%.

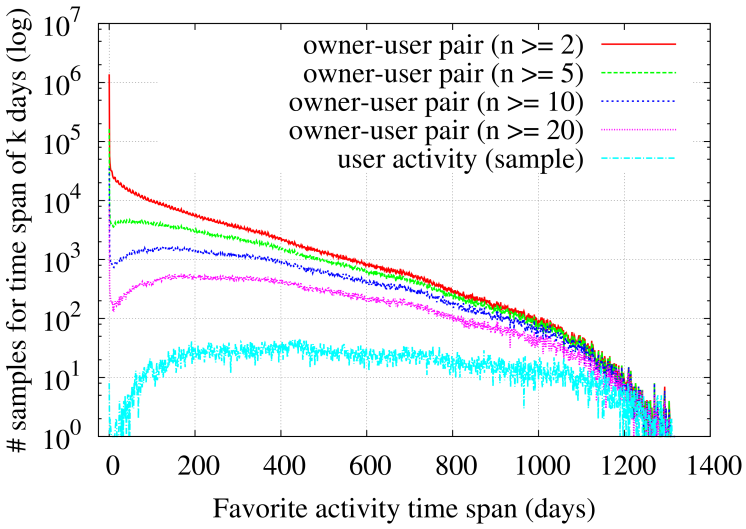


Figure 8.3: Time span of interaction between the owner of the photo and the user performing favorite actions.

Time Span of Owner-User Interactions

We created a histogram of favorite actions for all user-owner pairs in our data set. Note that with our notation, a user selects a photo as a favorite and an owner is the one who uploads that photo.

We analyzed only user-owner pairs with at least n favorites in total. As Fig. 8.3 shows, there are high peaks in very short time periods (less than a day). Note that we are considering only users that are connected by any of the relationship types, therefore the analysis shows that the favorite action happens in bursts and in many cases users do not return to favorite more photos of those owners: users tend to favorite photos of connections in short bursts.

Temporal Locality of User's Interests

We analyzed favorite actions that were close to each other in time and observed how many of the favorites shared a particular feature (*e.g.*, were uploaded by the same owner). In Fig. 8.4 we see that a large number of photos favorited in less than an hour are likely to be from the same owner, or group. Unexpectedly, in Fig. 8.4 we can observe subsequent daily peaks

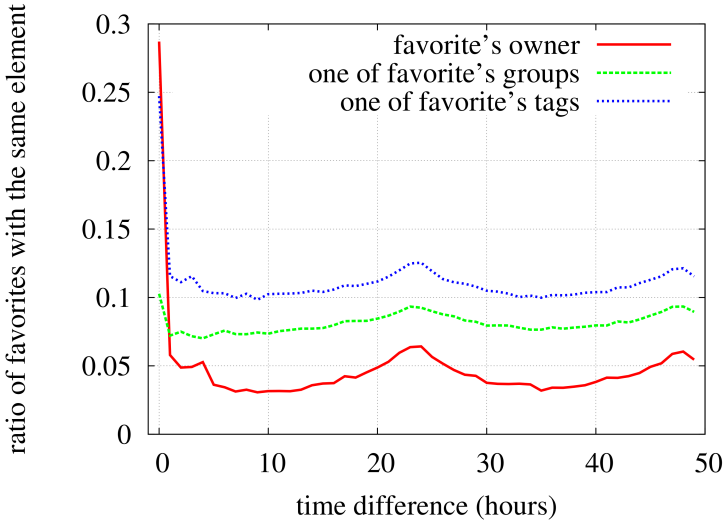


Figure 8.4: Likelihood of favoriting a photo with the same user, group or tags for a given (short) time period.

for owners and tags. In Fig. 8.5 similar peaks are observed for weeks. One interpretation of this is that when a user is interested in a picture with a certain feature, pictures that share this feature are more likely to be favorited.

8.4.4. Favorite Sessions

Another interesting aspect of favorite actions is their *burstiness*, in other words, measuring whether favorite actions occur uniformly over time or in bursts. We analyzed favorite actions within sessions, assuming that favorite actions are performed in the same session if the time difference between each pair of consecutive actions was lower than 30 minutes. Given this constraint, we measured the size of each session, comparing the two last groups described in Section 8.4.3: users with more than 200 favorites in total, and the $24K$ sampled users. The size of the sessions is much smaller for the sampled users, for which we considered only favorites performed within 10 hours of adding a photo.

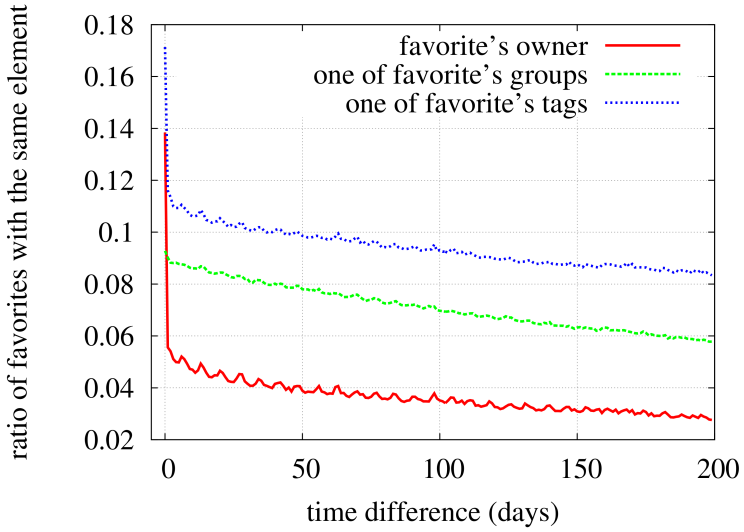


Figure 8.5: Likelihood of favoriting a photo with the same user, group or tags for a given (long) time period.

We found that approximately 70% of the sessions have favorited photos of no more than 3 different owners. In almost 21% of the sessions the photos selected as favorites are from a single owner, and in about 35% of the sessions from two owners.

8.5. Computational Features

Below we describe a number of features derived from the outcomes of the data analysis, and we perform a simple evaluation of their suitability in terms of how well they are able to recall favorited photos. We also build a baseline favorite recommender as a proof of concept.

Since we have information on which photos have been favorited, we perform the evaluation by assuming that we want to predict a particular favorite action. In other words, let's say that at time t a user favorites a photo p. When the user favorites that photo, he choses it from a set of photos S. In our analysis, we simply consider all photos in set S and examine which features might be more useful in predicting photo p, *i.e.*, the one that was selected as a favorite.

We will use the following notation:

- **Recipient** – a user who is receiving photo recommendations.
- **Owner** – the owner of the photo that is recommended to the recipient.
- **Recommendation event** – the moment in time (t) in which the system presents the recommendation to the recipient. We assume that the favorite action takes place when a photo is recommended (*i.e.*, the moment of the recommendation event is equal to the moment of the favorite action).
- **Search space** – the set of photos S that are considered for recommendation in a single recommendation event. In this analysis we focus on photos that were uploaded by *connections* of the recipient, at most 10 hours before the recommendation event.

8.5.1. Photo Based Features

Photo based features are extracted from each photo that is considered for recommendation (*i.e.*, each photo in the search space S defined above).

- **Photo recency** – timestamp of the upload of the photo (users are more likely to favorite recently uploaded photos, see Section 8.4).
- **Number of favorites** – the number of times a photo was favorited by other users prior to the recommendation event.
- **Number of comments** – could be indicative of interest in the photo.

8.5.2. Photo Owner Features

Owner based features are related to the owner of the photo that is considered for recommendation, so all photos uploaded by a single user (owner) have the same owner features.

- **Likelihood of favoriting owner's photo** – the number of times a photo from the owner was favorited, divided by the total number of owner's uploads.

- **Inverted batch size** – the inverted number of photos by the same owner in the search space. The assumption behind this feature is that users who tend to upload large sets of pictures are less concerned with their quality.

- **Recency of connection link** – timestamp of establishing a connection between the user and the owner: users might be more curious about photos of recent connections.

8.5.3. Feature Evaluation

In the following experiments we used the favorite actions of 100 users randomly chosen from the sample set of 24k users described in Section 8.4.3. Therefore, we considered only favorites done on photos from *connections*, uploaded at most 10 hours before the favorite action.

The objective is to rank all the photos that were uploaded within that time frame by the user's connections, so the favorited photo is in the top of the ranking. Each favorite action was considered a separate recommendation event. In this setting, the dataset contained $38,211$ recommendation events in which the total number of photos was $4,632,013$ (on average 121 photos per recommendation event). We split the data into training and test sets, where the first 80 favorites of each user correspond to the training set ($8,000$ recommendation events).

Since photos uploaded within 10 hours are considered, it is possible that the user may have already marked some of them as favorites. Given that photos cannot be favorited more than once by the same user, these were omitted (on average 1.8 photo per recommendation event). All feature values in the training and test sets were calculated on the information that was available, prior to the recommendation event (following the timestamps of user actions).

We used the average recall for the first k results as metric to determine the accuracy of features. Recall@k is the number of true positive instances among the first k results of the ranking, divided by the total number of positive instances. In each test case there is always one true positive instance (the favorited photo), and this measure represents the ratio of recommendation events for which the ranking was able to place the favorited photo among the first k photos of the search space.

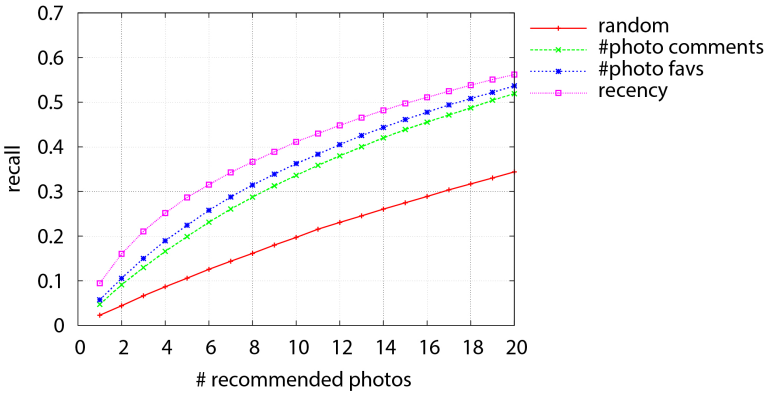


Figure 8.6: Accuracy of features in photo recommendation task using photo based features.

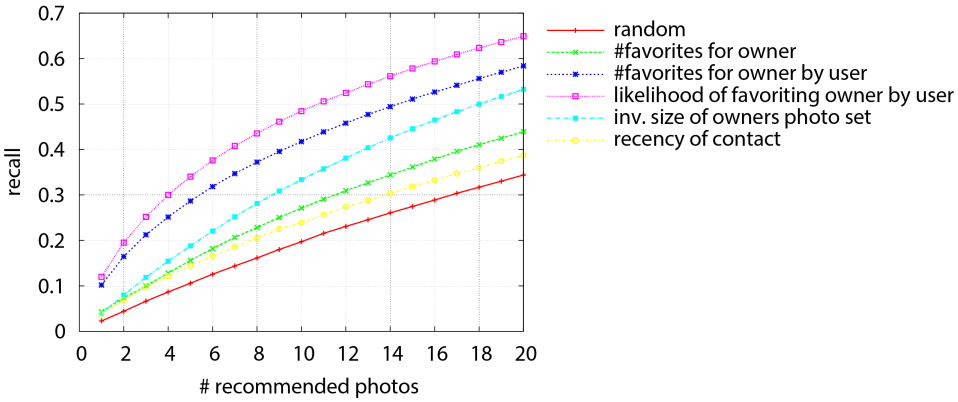


Figure 8.7: Accuracy of features in photo recommendation task using user based features.

Accuracy of Photo Based Features

Among the three photo based features (*i.e.*, *photo recency, number of favorites, number of comments*), the recency of the photo turns out to be the most accurate, see Figure 8.6 for the results. This feature is also the third most accurate among all tested features. The good performance of this feature can be predicted by observing the relation between the recency and the ratio of favorited photos.

However, it is also possible that the performance of the recency feature is biased by the fact that the most recent photos of a user are shown first in the Flickr interface.

The number of favorites and number of comments prior to the recommendation event represent actions by other users (*i.e.*, those that do not own the photo, and those that are not receiving the recommendation). Such actions are commonly used in standard recommendation techniques based on collaborative filtering. Both features have lower accuracy than the recency of a photo. Among the photos in the search space, 90% are already marked as favorites were favorited less than 10 times. This is expected given that we consider only photos that were uploaded at most ten hours before the recommendation event.

Accuracy of Owner Based Features

The main feature describing an owner of a photo is the number of favorites of his/her photos prior to the recommendation event. We tested three features: total number of favorites for the owner of the photos (#favorites per owner), total number of favorites marked by the recipient for the owner (#favorites for owner by recipient), and likelihood of owner's photo being favorited by a specific user given the photos in the search spaces of all recommendation events prior to the current one (likelihood of favoriting owner by user).

The first metric has the highest coverage; the last is most likely to have the best precision. Surprisingly, we can observe a very large difference between the total count of favorites and the personal count of favorites. The former has very low accuracy, which is unexpected.

The third feature is clearly superior. It measures the likelihood of an owner's photo being favorited by a recipient. The feature is personalized, which means that a separate likelihood value is calculated for each recipient. The high accuracy of this feature suggests that users tend to have a set of owners whose photos they frequently favorite. Indeed, in the sample, 40% of favorites are from owners who were favorited already five times (the total ratio of photos in the search space from these users is 13%). On the other hand, owners with no prior favorites contribute with 41% photos in the search space, but only 20% of favorites come from them.

The inverted size of owners' photos has reasonably good accuracy, suggesting that users who submit large sets of photos are in general, less likely to

submit interesting photos. The last user based feature – the recency of user connections has very low accuracy.

Accuracy of Similarity Based Features

In addition to photo and user-based features, we tested a range of features calculated based on the similarity of users and photos. We found that comparing to other features similarity between recipients and owners/photos has low favorite photo prediction accuracy. It appears that tags and groups are not as important in choosing favorited photos as who the photo owners are. Two additional reasons for low accuracy of similarity based metrics is the sparsity of tags and groups and the fact that users often assign the same set of tags to a large group of photos.

8.6. Discussion of Recommendation Results

We tested two recommendation strategies. The first strategy is to frame the recommendation task as a binary classification problem. In this case, the favorited pictures are treated as the only positive instances. The features of the photos are used to built classifiers. For each tested photo, we assign its ranking score as the confidence value obtained by the classifier in classifying a photo as positive instance. The second strategy is a simple linear combination of ranking scores proposed by the features. It utilizes the fact that most of the numeric features already convey some notion of the likelihood that a photo will be favorited. In all experiments we used the same dataset as was used for the evaluation of the accuracy of the features.

8.6.1. Classification

We tested a wide variety of classification algorithms, among them we found that Gradient Boosted Decision Trees (GBDT) approach presents the best performance, both in terms of effectiveness and efficiency. This result is in-line with the classification algorithm proposed for photo recommendation by van Zwol *et al.* [110]. In all presented experiments, we used the GBDT algorithm based on decision trees (REPTree algorithm) as weak learners. We have tested the algorithm in two settings. *Global* setting, in which the first 80 favorites from all users were used to train the global classifier ($8,000$ positive instances, $540,802$ negative instances). *Personal* setting, in which a separate classifier was built for each user, based on the 80 positive instances for a user and the corresponding negative instances. To capture the temporal

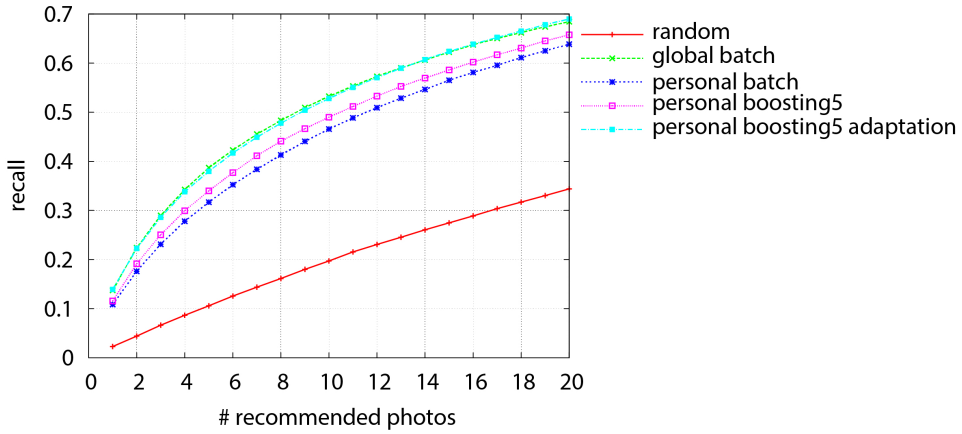


Figure 8.8: Classification.

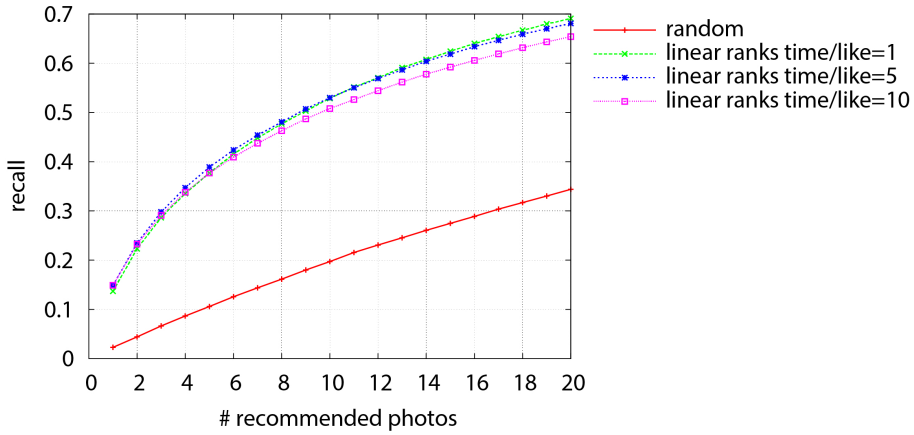


Figure 8.9: Linear combination.

characteristics of the data we used another *boosting* layer on top of GBDT. We built 80 weak classifiers for each set of 5 consecutive favorites. The final score of the boosting algorithm was the sum of scores of all weak learners. Finally, we run the boosting algorithm in *adaptive* mode, in which each newly added favorite (together with four preceding favorites) was used to build a classifier that replaced the oldest classifier in the list (sliding window approach).

Global classifier clearly outperforms the personal version of the basic classifier (Figure 8.8). It suggests that the number of instances used to train personal classifiers is insufficient. The performance of the personal classifier

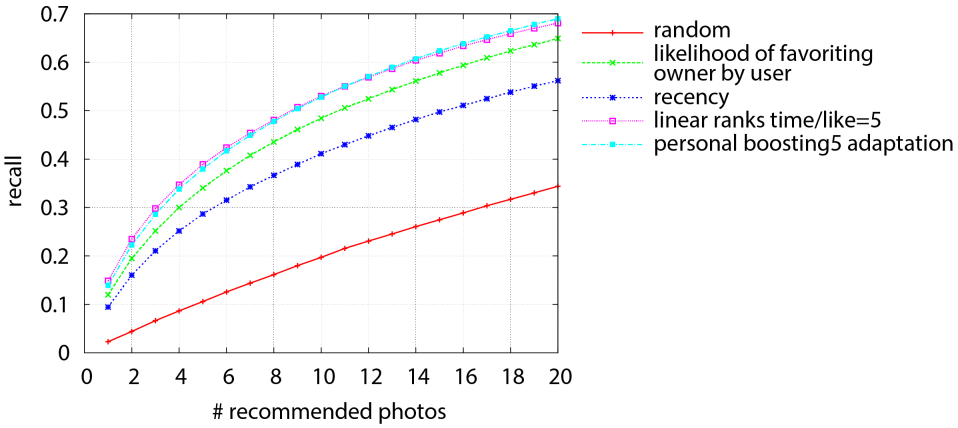


Figure 8.10: Comparison with feature baselines.

is slightly improved by the use of the additional boosting layer. However, only the adaptive version of the personal algorithm is able to perform equally well as the global classifier.

8.6.2. Ranking Merging

We decided to merge two of the most accurate features from photos and user based features, namely the recency of the photo and the likelihood of owner being favorited by a user. In each recommendation event the values of features were normalized for all photos in the search space. Later, they were linearly combined using a parameter α that defined their relative importance (Eq. 8.1). Despite its simplicity the recommender based on the linear combination of recency and owner likelihood values has comparable performance to the top classification based recommenders (Fig. 8.10). We have experimented with various values of α showing that putting more importance to the recency feature can improve the performance of the system for the top photos, but at the same time decreasing the accuracy for larger result sets (Fig. 8.9).

$$s_{sum} = s_{like} + \alpha s_{time} \tag{8.1}$$

8.6.3. Summary of Experiments

The results of our experiments show that even quite complex classification algorithms are not able to outperform basic linear combination of rankings from the two top features. It suggests that the discretization of features, that

is a necessary part of classification process, removes useful information from the feature ranking. At the current stage of work, it seems that combination of features is a more promising recommendation strategy. However, to test it further, we need a multiobjective optimization technique that would be able to discover the optimal merging parameters. The optimized function would be the average accuracy of the system for a set of recommendation events. The function is likely to be relatively easy to optimize using basic gradient descent like techniques. However, the computation of the function value would require the iteration over all photos in all recommendation events, which is likely to be computationally expensive.

8.7. Summary and Discussion

We presented an analysis of the favorite action in Flickr. The results show that users tend to favorite recent pictures of their connections, and in particular of their contacts; favorite actions tend to happen in bursts, particularly when considering individual user-owner pairs. For instance, it is common for a user who favorites a photo of a connection, to favorite several photos of that connection in a very short period of time. We also examined different features, both for owners and photos, in order to determine how useful they might be in a recommendation task. In particular, we used the results of the analysis to build a number of computational features and tested their suitability in determining which photographs may be marked as favorites. Our work contributes to gain insights into the "liking" behavior in social media (at least in the specific case of Flickr), and to inform strategies for recommending items users may like. We made a comparative analysis of different features, based on user and photo information, with the aim to recommend which photo the user will mark as favorite.

In the next chapter, we will see different ranking approaches based on explicit and implicit information, and among those; we will compare favorites and graph-based strategies.

Chapter 9

Implicit and Explicit Ranking Approaches

The last chapter of this thesis provides a comparison among browsing graph-based algorithms and commonly used approaches based on explicit feedback. Our analysis on the results reveals the different characteristics of the feedbacks and techniques applied.

In this chapter, we compare different rankings of images in Flickr and investigate the factors that affect each ranking, in order to perform image recommendation. In social media platforms, ranking of images depends on many factors such as the social interactions and the visibility of the images, both inside and outside those platforms. In this context, neither the application of standard ranking methods, nor the subtleties associated with taking into account the social interaction, and internal and external factors, are clearly understood. To this end, we use a large Flickr dataset and investigate these factors by performing an in-depth analysis of several ranking algorithms using both internal (*i.e.*, within Flickr) and external (*i.e.*, links from outside of Flickr) factors. We analyze rankings given by common metrics used in image retrieval, such as the number of favorites, and compare them with metrics based on page views, such as the View and the View Time. The results of this chapter were published in [100].

9.1. Introduction

Many social media platforms function as somewhat independent ecosystems, in which users carry out a number of social activities. Particularly, in Flickr, users can share content and participate in multiple communities by submitting their photos to groups, by joining groups, and by performing several types of actions over Flickr content such as comments, notes, and favorites. Thus, the way the content is consumed is strongly influenced by all of the different social navigation paths that lead to it: a photo on Flickr, for example, can be linked from a user's favorite photo collection, from several groups, galleries, and via other mechanisms, including the "external" web (*i.e.*, URLs outside of the Flickr domain such as blogs, news articles, *etc.*).

As more social media platforms emerge, one of the key questions is whether traditional ranking algorithms, that do not take the subtleties of navigation patterns driven by social connections into account, can be successful within those ecosystems. In particular, the problem we are interested in addressing is the general ranking of images in Flickr (*i.e.*, we would like to rank all of the images or a subset of them, in the order of importance). Such ranking can have many applications, including retrieval, and information discovery, among others.

The importance of images in Flickr, or of "nodes" in similar social media platforms, might depend on a number of *internal* and *external* factors. For example, an image that is very popular in a group with a cult following, may have been marked by many users as favorite image. The image might have received a large number of favorites due to its visibility in the specific communities. In contrast, an image of an important real world event (*e.g.*, the British Royal Wedding) might get a high number of views, not due to its visibility in specific communities, but due to its well connectedness by multiple external (*i.e.*, outside of the Flickr domain) media outlets, and get comparatively few favorite marks. One of the key questions is thus what the impact of those external and internal factors is on ranking and selection of content.

In this chapter, motivated by the scenario described above, we investigate the factors that affect image ranking by performing an in-depth analysis of the results of several ranking algorithms, taking into account both the internal (*i.e.*, within Flickr) and the external (*i.e.*, links from outside the Flickr domain) factors. In particular, we analyze rankings given by common metrics used in image retrieval (*e.g.*, number of favorites), and compare them with the metrics based on page views (*e.g.*, View, View Time). More

specifically, in order to take into account the structure of Flickr in terms of navigation paths to and from specific images, we represent the navigation patterns of the users with the *BrowseGraph*, and combine session models with some of these metrics. We implement PageRank and *BrowseRank*, and, compare them with different rankings.

Our main contributions can be summarized as follows:

- We compare five different implicit and explicit image ranking methods, evaluating them with a number of features that give us insights about which aspects each ranking method emphasizes.
- We introduce a variation of the *BrowseRank* algorithm, in which navigation patterns are used to assign a different damping factor to each node in the graph.
- We analyze the connectivity patterns of a large *BrowseGraph* extracted from Flickr. Results point to structural peculiarities that differentiate browsing graphs from other complex graphs like social and similarity networks.

To our knowledge, this is the most detailed comparison between image ranking algorithms in terms of number of baselines and features considered, and it is the first attempt to use *BrowseRank* for an image ranking task.

9.2. Ranking by BrowseGraph

In this section, we present an analysis of the Flickr *BrowseGraph* with the filtering and the referrer URL taxonomy respectively described in Chapter 3 (Section 3.3.1). Furthermore, we briefly describe the original *BrowseRank* algorithm [72] and the modified version adapted to our domain.

9.2.1. Analysis of the BrowseGraph

The *BrowseGraph* we extract from the Flickr sessions has about 50 Million nodes and 95 Million arcs, with a very low graph density of around $3.8 \cdot 10^{-8}$. Statistics on the average degree connectivity and the graph size for different Flickr entities are reported in Table 9.1. Higher in and out degree of group and user nodes compared to photos, suggests, as one might expect, that thematic groups and user profiles are hubs for the exploration of the website. The role of groups as navigation hubs is confirmed also by the inspection

	All	Photos	Groups	Users
#Nodes	49,275,691	46,569,946	183,996	2,521,749
$\langle k_{in} \rangle$	1.94	1.57	13.72	7.99
$\langle k_{out} \rangle$	1.94	1.51	13.69	9.05

Table 9.1: *BrowseGraph* statistics, with detail on single node categories, where $\langle k_{in/out} \rangle$ denote the average in- and out-degree.

	%Links			$\langle w \rangle$		
	Photos	**Groups**	**Users**	**Photos**	**Groups**	**Users**
Photos	0.6182	0.0098	0.1071	1.49	1.21	1.44
Groups	0.0114	0.0092	0.0057	1.54	1.44	1.65
Users	0.1332	0.0075	0.0979	1.48	1.41	1.32

Table 9.2: Flows and weights in the *BrowseGraph*. Cells report the overall percentage of links flowing from a node type to another and the average weight $\langle w \rangle$ of edges according to the type of the endpoints.

of the navigation flows between all of the possible pairs of node categories (Table 9.2), which shows that links from groups towards photos or users are on average used more often (i.e., have higher weight) than other link types. Moreover, it appears that groups and user pages attract traffic from many sessions, but soon redirect this traffic to particular photos. This can be inferred by the fact that the in-degree distribution for users and groups is heavier and broader than for photos, but the scenario is reversed when considering the distributions of the edge weights towards each node type (see Figure 9.1). In a nutshell, sessions end up in groups and user pages from anywhere in the network and from there they tend to converge to the most interesting or well positioned photos in the page.

Despite the important role of groups and user pages, the majority of arcs in the *BrowseGraph* are due to the navigation from one photo to another (62% of links, see Table 9.2). This is partially due to the disproportion in the cardinality of the three node categories (photo nodes account for 95% of nodes in the graph), but mainly it is the result of frequent navigation patterns. In fact, as shown in Figure 9.2, users very often browse photos of

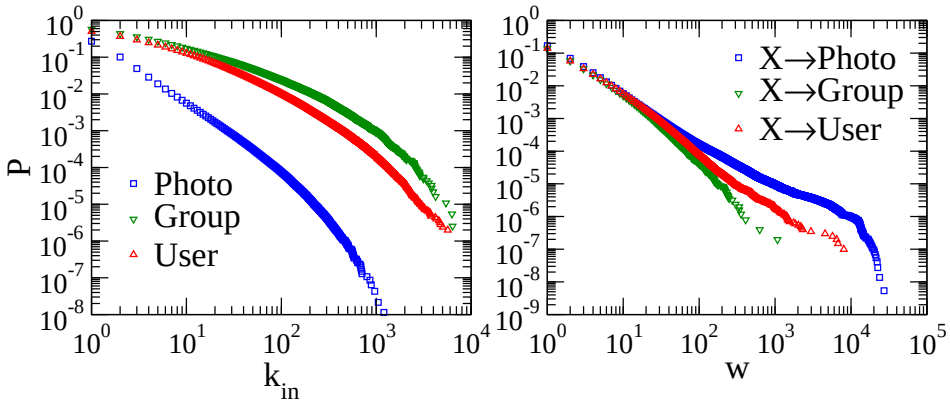


Figure 9.1: CCDF of the in-degree (k_{in}) for the three node types in the *BrowseGraph* (left). CCDF of arc weights for arcs terminating in nodes representing photos, groups or users (right).

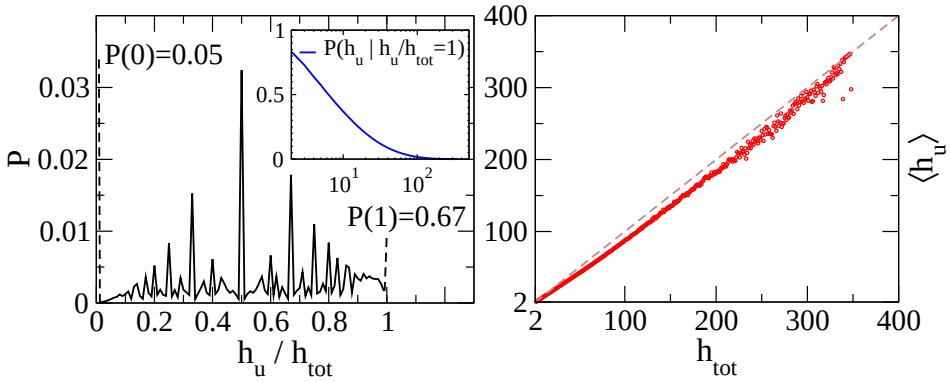


Figure 9.2: Distribution of ratio of session hops between two pictures belonging to the same owner (h_u) over the total number of hops h_{tot}. The inset shows the CCDF of the number of hops for the sessions visiting only nodes of a single user, which constitutes the majority of cases (left). Average number of hops between pictures of the same owner $\langle h_u \rangle$ at fixed session length (right). Points lying almost on the diagonal mean very high correlation.

the same owner and this happens not only for short sessions, as highlighted by the rather broad distribution of session length for this case. This behavior is largely determined by the Flickr *photostream*, that we analyzed in Chapter 6, which shows a strip of 5 photos from the same owner.

More generally, while surfing the Web, every user who visits a page eventually leaves following another link (or, more unlikely, end her session). As a result, a network created by the composition of such browsing patterns will have very high *balance* between in-and out-connectivity of nodes, as shown in Figure 9.3 (top). Such structural feature clearly differentiates navigation graphs from social graphs, in which popular individuals such as celebrities attract many connections and return a few back. The observed balance pattern gets slightly blurred only for very highly connected pages. Specifically, as shown in Figure 9.3 (bottom), well connected groups tend to have a higher in- than out-degree, while the distribution of the ratio of in-strength over out-strength for the most visited photos has a heavier tail towards values greater than one rather than towards zero. This confirms a scenario where the user navigation when not jumping from one photo to another, flows to hubs and gets redirected to popular photos.

Finally, a prevalent unidirectionality of browsing patterns can be evinced by the very low portion (0.17) of directed arcs $A \to B$ having a reciprocal $B \to A$, namely the reciprocation of the network. Again, this parameter is another footprint that discriminates navigation networks from social networks, which are on average highly reciprocated due to conventional social protocols.

9.2.2. Definition of BrowseRank

The *BrowseGraph* just outlined contains the information about user navigation paths and browsing behavior within Flickr. For example, the tendency to visit pictures in succession, moving directly from one photo to another, or exploiting the group and user nodes as hubs in order to select interesting photos and continue browsing.

Our goal is to use the computed *BrowseGraph* to rank entities inside Flickr. Since our *BrowseGraph* contains different types of nodes (photos, users, groups), not only photos are ranked. The obtained rank should capture well the global interest patterns leading the web surfers to any of the entities considered. In this work we consider the rank for the photo nodes only, but in principle the rank scores obtained for user and group nodes can be used as well for different tasks.

Relying on the *BrowseGraph* structure alone may lead to a series of problems. Due to the low density of the graph (see Section 9.2.1), the ranking could be biased towards nodes with high degree (*e.g.*, a user with a large number of photos or spammers), regardless of the quality of the entities. Moreover, important node attributes such as the time spent on them or

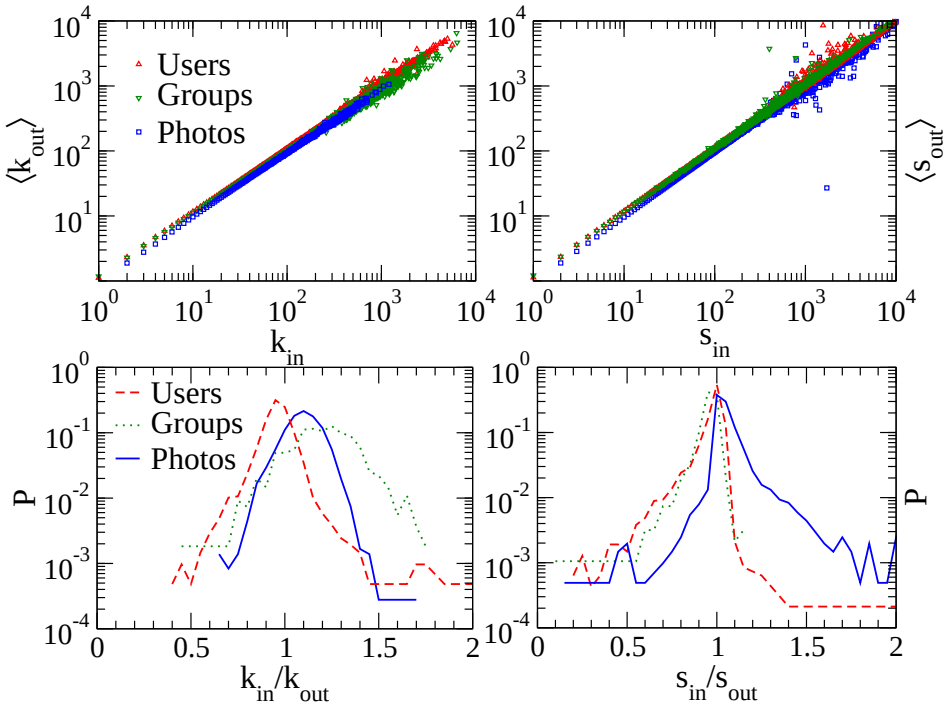


Figure 9.3: Average out-degree $\langle k_{out} \rangle$ and out-strength $\langle s_{out} \rangle$ at fixed values of in-degree k_{in} and in-strength s_{in}, for the three node types (top). Distribution of points almost perfectly aligned on the diagonal reveal the extremely high correlation between the amount of in and out session traffic which characterize navigation networks. Distribution of the ratio between in- and out-degree (in-and out-strength) for nodes with an in-degree (in-strength) higher than 500 (bottom). The different skews of the distributions highlights the different roles of the three node types in browsing.

popularity, would not be taken into account. The ranking needs therefore to be adjusted using additional information. We applied and improved an algorithm that takes into account the time spent by the user on a page and uses this information to readjust the values returned by PageRank.

BrowseRank [72], is a ranking algorithm based on a continuous time Markov process model that exploits the link structure of the *BrowseGraph*. As opposed to the classic Markov process, *BrowseRank* takes into account the time that users spend on the page. In the context of Flickr, time spent on a photo could be a good indicator of interest by the user. Next, we describe the algorithm and the way in which we adapted it to the Flickr

BrowseGraph.

Continuous-time Markov Model

As in [72], we use the Continuous-time Markov Model represented by the matrix $P(t) = [p_{nm}(t)]_{N \times N}$, where p_{nm} represents the transition probability from vertex v_n to v_m for time interval t.

The *BrowseRank* algorithm computes the stationary probability distribution $\{\pi_i\}$ by using the *transition rate matrix* $Q = [\frac{\partial}{\partial t} P(t)](0)$ and the Embedded Markov Chain (EMC). The EMC is a discrete Markov process derived from Q (for details see [72]). Given the stationary probability distribution of the EMC $\tilde{\pi}_i$, we can compute π_i using

$$\pi_i = \frac{\frac{\tilde{\pi}_i}{q_{ii}}}{\sum_{\{v_j\}} \frac{\tilde{\pi}_j}{q_{jj}}} \ . \tag{9.1}$$

Embedded Markov Chain

The EMC is a Markov Chain whose transition probabilities are based solely on the observed transitions between entities in the *BrowseGraph*:

$$G = \langle \{v_i\}, \{e_{ij}\}, \{w_{ik}\} \rangle \ , \tag{9.2}$$

where $\{v_i\}$ is the set of vertexes, $\{e_{ij}\}$ is the set of edges and $\{w_{ij}\}$ the set of weights associated with the edges.

In addition, for each node j, we compute the *reset probability* σ_j, *i.e.*, the probability of starting a new session in j as the number of sessions that start in j over the total number of sessions. Moreover, for each node j, we compute the *stop probability* α_j, *i.e.*, the probability of ending the session in j as the number of sessions that end in j over the total number of sessions that contain j. Both probabilities have been smoothed in order to avoid zero probabilities.

The transition probabilities of the EMC are computed in the following way:

$$emc_{ij} = \alpha_i \frac{w_{ij}}{\sum_{\{v_k\}} w_{ik}} + (1 - \alpha_i)\sigma_j \ . \tag{9.3}$$

Intuitively, Equation 9.3 indicates that as the user traverses node i of the graph, she may continue the navigation with probability α_i or randomly reset to any other node with probability $(1 - \alpha_i)$. In case she continues,

the transition probability is computed based on the observed transitions. In case she resets, the probability of ending up in node j is the reset probability σ_j. Equation 9.3 looks similar to the weighted PageRank algorithm [120], but we are able to exploit additional information that is not available to web crawlers. By having the number of sessions starting and stopping in a given node, we are able to estimate the specific reset and stop probabilities σ_i and α_i for every page i. The estimation of these parameters makes the random walk more realistic since it models the navigation of the user in a more accurate way. Equation 9.3 differs also from Equation 8 in [72] in the fact that we are not only estimating the reset probability, but also the stop probability. The additional advantage of this parameter estimation is that it avoids to manually set any parameter prior to running of the algorithm.

After computing the EMC transition probabilities, we compute the stationary probabilities $\{\tilde{\pi}_i\}$. Up to this point we have not taken into account the time spent by the user on the entities. It is indeed interesting to compare the performance of the ranking with and without this information. We will therefore save the $\{\tilde{\pi}_i\}$ and we will refer to them simply as the *PageRank*.

BrowseRank

As a final step, we include the information about the time spent by the user on the entities to improve the result of the previous section. For each vertex of the *BrowseGraph* v_i we compute the duration of the visits of users as follows:

- For each pageview p_n with timestamp t_n belonging to session s, we compute its duration d_n as the difference between the timestamp of the next pageview and its timestamp $d_n = t_{n+1} - t_n$. As we are not able to compute the duration of the last action of a session, we decided to discard it.
- We then compute the *aggregate durations* by summing up the duration of consecutive pageviews that refer to the same *BrowseGraph* vertex v_i.
- Finally, for each v_i we compute the sample mean $\bar{Z}_i$ and the sample variance S_i^2 of its aggregate durations.

We apply the additive noise model [72] to cope with noise deriving from different connection speeds and we compute q_{ii} by solving the optimization problem in the following equation:

$$\begin{aligned} \min_{q_{ii}} \quad & \left(\left(\bar{Z} + \frac{1}{q_{ii}}\right) - \frac{1}{2}\left(S^2 - \frac{1}{q_{ii}^2}\right)\right)^2 \\ s.t. \quad & q_{ii} < 0 \end{aligned} \tag{9.4}$$

We can now solve Equation 9.1 to compute the value of *BrowseRank* for every node.

The *BrowseRank* algorithm is straightforward to parallelize in Map-Reduce. In terms of complexity, the most demanding step is the computation of the stationary probability distribution of the EMC $\tilde{\pi}_i$. Using the power method, the overall complexity of the algorithm is $O(N \log(1/\epsilon))$, with N number of edges in the graph and ϵ a given degree of precision [12].

9.3. Evaluation

We compare the top 1,000 Flickr photos ranked using five different importance scores, specifically:

- **Favorites**: absolute number of favorite marks assigned to a photo. Favorites can be assigned only by Flickr users.
- **Views**: absolute number of views of the photo page (this includes users that are not logged in).
- **View Time**: cumulative time spent by all of the visitors of a photo page.
- **PageRank**: PageRank score of the photo page, with estimated start and stop probabilities as denumber of viewscribed in Section 9.2.2.
- **BrowseRank**: *BrowseRank* score of the photo page, with estimated start and stop probabilities as presented in Section 9.2.2.

The selected methods include a fairly general selection of explicit (Favorites), implicit (Views, View Time) and centrality-based ranking techniques (Page/BrowseRank). The number of favorites has often been used as an evaluation baseline in Flickr photo ranking [84] as it is the most explicit

indication of preference and the scores can easily be aggregated. Views and View Time are also often used for ranking in photo sharing sites due to the ease of computation. Although quantitative correlations have been found between the visit count and the explicit user feedback on photos [65, 85], we show that all metrics behave in appreciably different ways.

9.3.1. Popularity, Interestingness, and Diversity

When comparing different picture sets, image quality is just one of the parameters. In particular, when images are embedded in dynamic social environments, the interest people have in particular photos can be determined (or influenced) more by the social dynamics of a community (*e.g.*, a group in Flickr) than by the inherent quality of the photos themselves. Similarly, interest can originate externally (*i.e.*, many photos in Flickr are linked from outside of Flickr) and thus be important independently of their aesthetic qualities (*e.g.*, photos of important events).

Given that several factors can be taken into account in considering a ranking of images, we identified four importance macro-notions and we list some quantitative *features* for each of them. All of the features were then used as evaluation parameters to compare the rankings.

- **Internal popularity**. Popularity of a photo inside the Flickr community. Popularity does not necessarily imply quality, but directly expresses the interest of users in a particular item. Features describing photo popularity are the number of *Comments* the picture receives and the number of internal Flickr *Groups* in which it appears.[1]

- **External popularity**. We consider measures of external popularity: the number of search results obtained from a Google search (*Google Results*) using the photo page URL as a query, the *Google PageRank*[2] of that URL, and the number of browsing sessions originating from an *external URL* that visit the photo page as the first Flickr page.[3]

[1] Although placing an image in multiple groups does not automatically make it popular, one can argue that photos that appear in multiple groups can be considered to be more popular because they have wider exposure

[2] We obtained the Google PageRank querying the API for each photo page URL `http://api.exslim.net/docs/pagerank`

[3] For several queries, the Google search results were similar to those obtained by other search engines, so we used them as a representative metric

- **Collective attention**. Users not logged into Flickr as well as Flickr users who do not actively give feedback on photos, implicitly express their interest in specific photos by visiting the pages that contain them and by spending time on them. Therefore, we use the total number of views of a photo (*View*) and the cumulative time spent on the photo (*View Time*) as an aggregate measure of attention that a generic web user, whether or not logged into Flickr, devotes to that image.

- **Diversity**. One of the applications of ranking a large-set of photos might be to display the most interesting ones. In this case, a very homogeneous set of pictures may result appealing to some user categories but are less likely to attract a wide public. Assuming that photos belonging to the same user are on average more homogeneous than pictures taken from different users, diversity can be estimated by the number of different *photo owners*. Additionally, an analysis on the diversity of the corpus of *tags* of the photos can be a measure of the variety of concepts represented.

We use Views and View Time as both ranking metrics and evaluation parameters to draw a more complete analysis of other rankings. We could have done the same for the number of comments, but we omitted to use it as a ranking metric because its performance was very similar to the Favorites. Cumulative values of each of the features defined are shown in Figure 9.4. To give a long-range overview of the behavior of the different ranking strategies, we show the feature values for the top $1K$ photos. Nevertheless, since many applications need much shorter ranked lists, we report that the relative position of the different curves is nearly unchanged for the top 20 and top 100 photos, for all the metrics considered.

Results reveal that most Favorites have good internal popularity, being the top metric in both the number of groups and comments, but behave worse than any other metric in terms of external popularity and collective attention. In contrast, photos with top *BrowseRank* scores are less popular internally (even though their scores are comparable to favorites up to the top 100) but they attract relatively more collective attention and position above any other metric when counting external relevance and owner diversity. PageRank behaves worse than *BrowseRank* except for collective attention. Finally, Views and View Rime perform reasonably well for external popularity and by definition in collective attention, but surprisingly the ranked photos have relatively little popularity in groups and receive few comments.

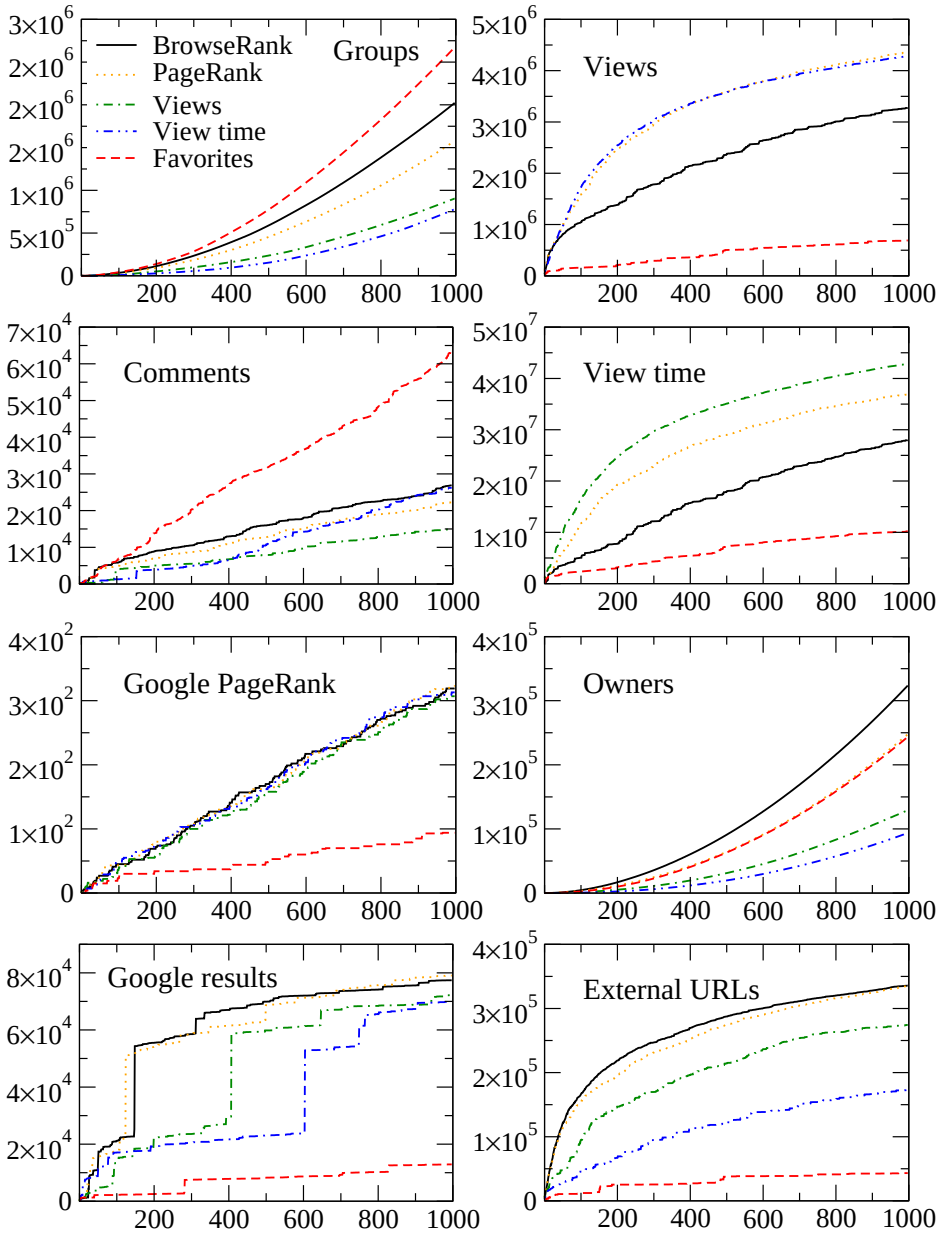


Figure 9.4: Comparison of the five ranking methods considered (*BrowseGraph*, PageRank, Views, View Time, Number of Favorites), according to eight features. Curves show the cumulative value of the features up to the top $N \in [1, 1000]$ results in the ranking. Views and View Time are used as both ranking methods and features.

	$\frac{\lvert photos_{tag}\rvert}{\lvert photos\rvert}$	$\lvert tags\rvert$	$\lvert set(tags)\rvert$	$\langle tags\rangle$	H
BrowseRank	0.73	**7,913**	**4347**	**7.93**	**11.23**
PageRank	0.75	7,129	3,583	7.39	10.57
Favorites	0.53	4,164	2,936	5.98	10.81
View Time	0.80	6,192	2,245	6.20	9.31
Views	**0.83**	6,523	2,113	7.14	7.14

Table 9.3: Statistics on the tag diversity for the top 1000 photos in the rankings. Columns report, from left to right: fraction of tagged photos, number of tags, number of distinct tags, average number of tags per photo, and entropy H associated to the tag frequency distribution. Entropy is given in number of bits ($\log_2$). Highest values are highlighted in bold.

Diversity in terms of tag categories is explored separately in Table 9.3. The richness of the annotation corpora from the five rankings are evaluated in terms of number of (distinct) tags appearing in the corpus or on single photos. Furthermore, we computed the entropy on the tag frequency distribution as a measure of uncertainty of the type of tags attached to a randomly selected photo. *BrowseRank* clearly outperforms all other metrics.

9.3.2. Quality from Visual Inspection

Assessing the quality of photos by visually inspecting them is a challenging task due to the intrinsic subjective component of the evaluation. However, to gain insights into how different quantitative features impact the type of images shown, we show the top 10 images for all of the 5 ranks considered (Figure 9.5).

Albeit any manual classification is ultimately arbitrary, we partition the photos in four well-recognizable, high-level categories that help to better understand the nature of the top photos. The pictures shown are assigned to one of the following categories: 1) *artistic* high-quality landscapes or portraits, 2) major natural and social *events*, 3) part of specific photo *series* or serial events, and 4) *peculiar* or curious shoots. The classification of each image is reported in Table 9.4.

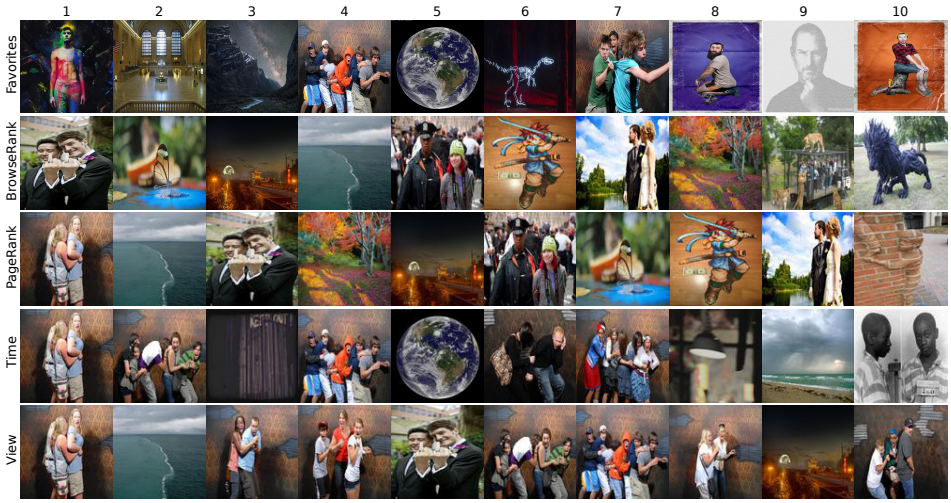


Figure 9.5: Top 10 photos for the five ranking strategies considered. Pictures include: (F2) shot of an empty railroad station during a hurricane in US, (F4 and similar) pictures of visitors to a horror house, (F8,F10) fun calendar series, (F9) memorial potrait of Steve Jobs, (B1) portrait in support of gay marriage, (B4) rare natural phenomenon of water masses at different densities melting one into another (the photo was broadcast by several news media), (B5) arrests during the "Occupy Wall Street" movement demonstrations, (B6) mosaics of a popular electronic-game character, part of a wider series, (B9) close lion encounters tourist van, (B10,P10) art installations, (T10) mugshot of the youngest African-American sentenced to death in the US, and (F1,B2, and more) artistic portraits, landscapes or photoart.

At first glance, Views and View Time rankings are dominated by a majority of photos depicting scared visitors to a horror house.[4] Traces of the same series, plus a couple of pictures from a humorous calendar series are present also among the top Favorites; besides that, artistic pictures are prevalent, followed by two photos related to breaking news. *BrowseRank* and PageRank have an almost identical set of pictures, in different order. They both contain as many artistic images as Favorites but more images related to trending topics or natural events. Series-related pictures are present (*i.e.*, horror house and mosaics of electronic games characters) but just as singletons. Photos of peculiar art installation or entertainment activities complete the ranking.

[4]See http://www.nightmaresfearfactory.com/

	Art	Events	Series	Peculiar
BrowseRank	2,3,7,8	1,4,5	6	9,10
PageRank	4,5,7,9	2,3,6	1,8	10
Favorites	1,3,5,6	2,9	4,7,8,10	-
View Time	5,9	10	1,2,3,4,6,7,8	-
Views	9	2,5	1,3,4,6,7,8,10	-

Table 9.4: Manual classification of top 10 ranked photos into four categories representing high-quality artistic images, natural and social events, picture series, and peculiar or fun images. Image numbers refer to Figure 9.5.

9.3.3. Analysis of the Results

The overall scenario emerging from the comparison shows that different metrics promote different types of photos. Rankings based on explicit feedback (*i.e.*, Favorites), boost pictures that are well spread across Flickr groups and that receive attention from active Flickr users, but that may not have great impact outside of Flickr. Top rated images tend to belong to a small set of owners, conveying a lower semantic variety than the pictures from centrality-based rankings, where artistic photos made by professionals are prevalent.

BrowseRank and PageRank, instead, overshadow a bit the very popular content *inside* Flickr to provide images with higher semantic variety and with apparently stronger interest from a broader part of the Web (outside of Flickr). This includes popular photos on trendy social events or pictures about popular fun facts or peculiar subjects. A positive side-effect of this is that photos that are related to popular *memes*, just inside Flickr (*e.g.*, horror house pictures), are downgraded and tend to disappear from the top ranking. Moreover, being based on the data from the navigation log only, centrality rankings are fully *implicit*. They do not need an active user commenting or voting the images. This means that *BrowseRank* and PageRank, are effectively more able to pick up diverse image collections, and produced more balanced lists by considering external links to the photos. Such algorithms can be profitably parallelized, making their computation efficient, even for big social media sites like Flickr.

Simpler metrics such as Views and View Time has the advantage of an easy computation, but overall, they perform poorly compared to others, at least in terms of diversity of the results.

9.4. Summary and Discussion

The problem of general ranking of images, in social photo sharing services, has not a widely-accepted solution, moreover, differences between different strategies have not been explored in depth so far. To shed light on this matter, we compared five possible ranking strategies in Flickr: explicit feedback (number of favorites), implicit user information (views and view time), and graph-centrality methods (PageRank and *BrowseRank*) applied to the *BrowseGraph*. In particular, we contribute to the definition of a customized version of the *BrowseGraph* that is limited to the navigation patterns inside the boundaries of the considered service, but that takes into account also the entry points of users navigating to Flickr from other domains. The purpose of such model is to express the complexity of navigation patterns in a meaningful way, which captures the importance that images have outside of the social media platform being considered. Unlike previous work in PageRank-based algorithms, we estimate a different damping factor for each page from the user session information.

A comparison between rankings was performed on a large Flickr dataset along several axes, including the internal and external popularity of ranked images, the overall attention that they attract from web users, their diversity in terms of ownership and semantic categories, and their visual appearance. Favorite-based ranking boosts mainly professional artistic photos that are very popular inside Flickr, but they are limited in variety and have low impact on the external Web. On the contrary, centrality-based methods, *BrowseRank* in particular, promote images that have attracted interest of external Web services like news media and produce more diverse rankings, minimizing the noise due to serial but relatively uninteresting photos periodically popping out in Flickr.

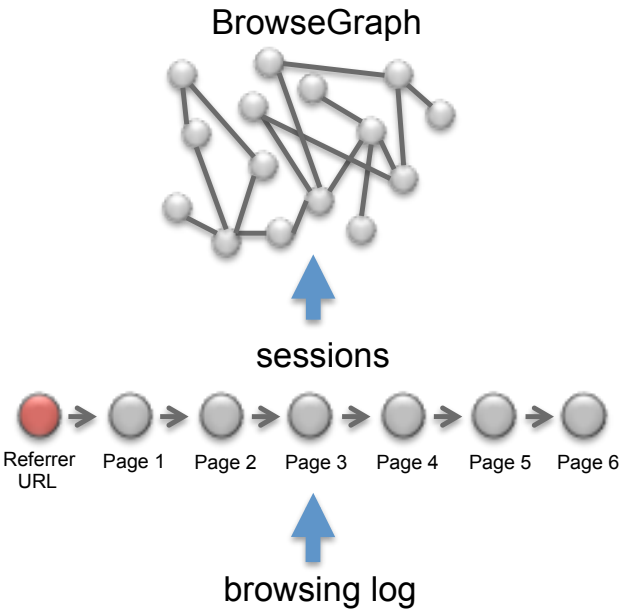
Chapter 10

Conclusions and Future Work

In this dissertation we perform different analysis and experiments regarding the users' browsing behavior. The browsing log is a very important data source for any service provider for two main reasons: it is always available, and it collects users' implicit feedback. It contains extremely meaningful information about how the user behaves within the website. Figure 10.1 summarizes some of the dimensions available in this type of log. The green checks mark the dimensions that we used in the experiments described in this thesis. We studied in depth some of these dimensions, in particular the referrer URL, that is a source of information still very undervalued in the literature.

However, the main drawback of the browsing log is the noise that characterizes it. It collects events that are generated by the navigation of the user that often need to be interpreted. This is one of the reasons why this data source has been exploited significantly less than explicit feedbacks, in particular in the domain of recommender systems. Understanding and exploiting the browsing log could be very challenging, and this is an additional motivation that leaded us to investigate it in this thesis.

The following sections summarize the main results, answer to the research questions drawn up in Section 1.1, and in the last part, discuss some possible research lines for future work.



IP Address	Bcookie	User Agent	Timestamp	Referrer URL	Current URL	More HTTP Data	Custom Data
✗	✓	✓	✓	✓	✓	✗	✗

Figure 10.1: Browsing logs features used in the experiments of this thesis, first to extract the sessions and then to build the *BrowseGraph*. The *custom data* contains a set of additional information that depend on the configuration of the web server. For example, event clicks in Javascript that do not need to refresh the page, social media sharing buttons clicks, and so on.

10.1. Main Results

This thesis explores in depth how to exploit the user browsing behavior, and in particular the referrer URL, in order to understand the preferences of the user and perform personalized recommendations. First, we analyze the relation between the referrer URL and the type of session the user does (discussed in Chapter 4). We find that the domain visited by the user before entering in the website, could be extremely meaningful in order to characterize the session of the user. We show, for the first time, that exist a relation between the referrer URL and the content that the users consume in their session. For example, in Flickr, users that land in the website from a mail

Referrer Domain	Page Layouts Contained in the Cluster
Aggregator	*consuming fresh content recently posted*
Mail	*managing contacts, such as adding/removing friends*
News	*browsing entire groups of photos*
Search	*searching with Flickr interface, mainly CC photos*

Table 10.1: Examples of the relations between the referrer URL and some clusters of users' sessions.

account, are most likely to manage the contacts of their social network and navigate through their contents. Users coming from news domains instead, tend to browse group pages, since they might contain photos related to the same topics of the news articles. Some of the outcomes pointed out by by our experiments are summarized in Table 10.1. These findings are extremely significant for a website that can adapt the layout of the web pages and recommend specific content to each user, with respect to the general interest of other users coming from the same domain.

Then, in Chapter 5, we exploit a browsing graph based on the navigational sessions of the users, called *BrowseGraph*, as well as browsing graphs grouped by the same referrer domains (*ReferrerGraphs*). We build and analyze these graphs collecting navigation patterns of users coming from *social networks* (Facebook, Twitter, Reddit), and from *search engines* (Yahoo, Google, Bing). We show the importance of the referrer URL for facing cold-start problems of newcomers in Chapter 7, and how the *ReferrerGraphs* help to solve recommendation tasks. However, there are cases where the external referrer URL is not available in the browsing log, and thus we cannot always apply our approaches. This happen, for example, when the users enter in the website through links shared in mail clients or social applications (*e.g.*, Twitter Applications). Due to the importance of the referrer URL that we acknowledge in this thesis, we perform some experiments with the aim to identify, through the session of the users, from which referrer domain they entered in the website. The result of our experiment, described in Chapter 5, shows how it is possible to identify the correct referrer domain just after few steps (*i.e.*, web pages visited by the user), filling the missing information and, as a consequence, permitting to personalize the content also for these users. Moreover, since we use centrality-based algorithms to estimate the

	Referrer Domain	Precision@1	MRR@3
social	Facebook	0.43	0.50
	Twitter	0.47	0.54
	Reddit	0.48	0.54
search	Google	0.24	0.30
	Yahoo	0.22	0.27
	Bing	0.27	0.34

Table 10.2: Summary of the results of the news article recommender system based on the *ReferrerGraph* with the mix-edge combination (see Chapter 7 for details).

importance of the web pages to compute a general ranking of items. We also experiment some limitation of PageRank in order to validate its reliability on these types of graph.

Given these analysis, in Chapter 6 and Chapter 7, we implement different recommender systems based on the browsing patterns of the users, and more specifically on the *ReferrerGraphs*, and we evaluated them with Flickr and Yahoo News data. These applications, face the cold-start problem of newcomers, a problem extremely important for any service provider that wants to increase the engagement of the new users, to present them interesting and novel content. Particularly, about the news domain, we have shown how, exploiting the referrer domain associated to the current navigation of the newcomer, it is possible to predict the next page the user is going to visit. Table 10.2 shows the main results for one of the algorithms that we proposed, based on *ReferrerGraphs*. It highlights the good accuracy, in terms of Precision and MRR, of the recommender for 6 referrer domains that we considered. In particular, for the social network domains, the first article recommended is in more than 40% of the cases the one the user will consume. This result is obtained only exploiting the browsing log, in particular the historical users sessions and the referrer URL from where the users were coming. For a news website, being able to capture the interest of newcomers at the first visit, results in increasing the overall flow of users. This is in line with the goals of the majority of on-line free service providers: increase the amount of new visitors and boost the volume of traffic. The approaches that we described, can be applied in any domain and by any service provider.

Evaluation Features		Favorites	PageRank	BrowseRank
External Popularity	Google PR	3	**1**	**1**
	Google Results	3	2	**1**
	External URLs	3	2	**1**
Internal Popularity	Groups	**1**	3	2
	Comments	**1**	3	2
Collective Attention	Views	3	**1**	2
	View Time	3	**1**	2
Diversity (summary)	Owners	3	**1**	**1**
	Tag Statistics	3	2	**1**

Table 10.3: Summary of the Flickr image ranking evaluation among the centrality-based approaches based on *BrowseGraph*, and the more standard based on favorites. The number represent the position among the 3 algorithms with respect to the evaluation features (see Chapter 9 for details).

The final part of this thesis presents first, in Chapter 8, an analysis of the users' behavior with explicit feedback, such as favorites, in the Flickr photo-sharing website. Then, it compares different explicit and implicit approaches in order to rank web pages that represent the content of the website, such as photo pages in Flickr. In Chapter 9, we compare graph-based solutions that exploit the *BrowseGraph* and the referrer domains. These latter ones are included in the *BrowseGraph* as external nodes, allowing the ranking approach to consider also the external impact of the images that belong to Flickr. A small summary of the results is shown in Table 10.3. The contribution of the referrer allows the graph-based ranking to obtain different and more interesting images for a more general audience, *i.e.*, users that are less involved in the Flickr network such as passive users, non-registered visitors or newcomers. Moreover, the images ranked by these graph-based algorithms, have a stronger external popularity in terms of search engine results (reachability), PageRank (general importance), and referrer URLs where they are shared (spreading).

In this dissertation we studied in depth the browsing log, mainly constructing the *BrowseGraph* and the *ReferrerGraphs*. Our experiments shown how these sources of implicit information help to understand the users' prefer-

ences and can be used for ranking and recommendation. We believe that the contributions presented in this thesis will motivate more research around these browsing graphs, and in particular, on the referrer URL, since we shown their big predictive potential. At any rate, we hope the findings highlighted in this thesis will lead to a greater consideration of these sources of information.

10.2. Detailed Results

In this section we answer the research questions drawn up in Chapter 1.

Personalize Content for Newcomers

Q1. *How could the content of interest be promoted to the users if we do not know anything about them? Is it possible to perform a personalization of content to newcomers?*

We find that the referrer URL is correlated with the type of session performed by the users. In other words, when the user enters the website, the external URL from where the user is coming from, contains information that shed light on the type of content the user is consuming in the website. As a consequence, it is possible to estimate the interest of the user by considering the referrer URL. Moreover, if we collect the sessions of the users in the *BrowseGraph* then we can compare the current navigation of a new user (*i.e.*, newcomer) with the history of previous users' sessions. This leads to an understanding of the current user's taste. We show how to implement a collaborative filtering recommender based on this finding that results to be very accurate in predicting the content the user is going to consume.

BrowseGraph and *ReferrerGraphs*

Q2. *How do PageRank-like algorithms behave on the BrowseGraph and on the ReferrerGraphs? Are these graphs reliable in this context to understand user behavior?*

The *BrowseGraphs* and the *ReferrerGraphs* have not been well studied in the literature since they were proposed in 2008 [72]. We find that they suffer of the Local Ranking Problem but that is possible to reduce the error regarding

the local PageRank scores, by expanding the graph with a set of neighbors' nodes (*i.e.*, web pages). We use these findings to estimate the importance of the images in Flickr in order to perform a ranking of images based on the users' navigation patterns. In addition, we show how these graphs are extremely useful to identify the intent of the newcomers by exploiting the browsing patterns of previous users.

BrowseGraph for Recommendation

Q3. *Would it be possible to exploit these browsing graphs in order to recommend novel and interesting content to users? What is the contribution of these graphs compared to standard methods based on implicit or explicit data?*

The graphs based on browsing logs contribute to face the cold-start problem recommending interesting content to newcomers. We implement algorithms based on the *BrowseGraph* to rank Flickr images, and we compare these graph-based approaches with more standard methods based on favorites, views and time spent. The graph-based approaches that consider the referrer URLs of the users' sessions, lead to a different ranking compared to the other methods. For example, they rank higher images that have a stronger *external impact* (*i.e.*, outside Flickr), in other terms, Flickr images that are very popular on the Web but that do not have many favorites or visibility inside Flickr itself. This outcome highlights how, using the *BrowseGraph*, it is possible to get information about the users exploiting their browsing behavior. Our experiments on news recommendation exploiting these graphs, result to be very accurate especially for users coming from social networks such as Twitter, Facebook and Reddit.

10.3. Future Work

The directions in which the work explained in this thesis can be expanded and applied in related domains, are discussed in the following paragraphs.

Exploit Remaining Features of the Browsing Log

Figure 10.1 shows the features that we leveraged in this thesis to build the sessions of the users. However, there are other features that could be exploited to increase the information regarding the user. The IP address for

example, tells us the location from where the user is connecting, in other words it allows the geo-location of the users that might lead to different investigation and personalization location-oriented. Other information are contained in the *custom data*, that it strictly dependent on the configuration of the web server. It might contains events based on Javascript or buttons clicks. For example the sharing of certain web pages through *social buttons* on websites such as Twitter or Facebook. This information is not always available since it depends on the ad-hoc configuration of the server. The actions performed by the users when they share on-line content (*e.g.*, images, news articles, videos) through social platforms, have been found to be extremely related to the overall consumption of the item [23]. These information, when they are available, increase the knowledge about the user's interest, especially for the active users. As a consequence the recommender system will gain in accuracy.

Extending the information at our disposal, might lead to an improvement of the accuracy of the recommender, or even to a recommendation of different types of content (*e.g.*, items for which the location is a determinant feature).

Integrate User and Item Profile

The user profile has not been considered in this dissertation, since we focused on the browsing log and on the referrer URL. However, collecting the preferences of the user and integrating them with these implicit activities, significantly extends the overall information available regarding the user. The recommender system presented in Chapter 7 based purely on the *ReferrerGraphs*, reaches very good levels of accuracy only with implicit feedback, as summarized in Table 10.2, especially for certain types of referrers (*e.g.*, social network). It deals with the cold-start problem of newcomers, but in the case of known users where the service provider has information about their previous activity, it is possible to improve the news articles recommended by filtering the content that fits the known preferences of the users.

Similarly, collecting the information about the items that we intend to recommend and integrate these profiles with the implicit data, will increase the details of the information at our disposal, facilitating its understanding and use. For example, in multimedia domains such as Flickr, the algorithms presented (see Chapter 6) could be greatly enhanced by taking into account more content features, such as EXIF data, comments, and visual features.

Continuous-Time Models

In terms of modeling, the recommender systems proposed in Chapter 7 could be extended using continuous-time models. We built the *ReferrerGraphs* at hourly time intervals, where the recommendation at time slot t where done using the *ReferrerGraph* of time $t-1$. Even if we obtained a very good accuracy that is summarized in Table 10.2, our approach cannot recommend news articles that are published at the same time slot t. Extending our approach for a real-time application will improve the precision of the recommender for these items (*e.g.*, news articles), since a continuous-time model will not be limited by hourly time slot.

Bibliography

Each reference indicates at the end the pages where it appears.

[1] G. Adomavicius and a. Tuzhilin. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, June 2005. 17

[2] D. Agarwal and B.-C. Chen. flda: Matrix factorization through latent dirichlet allocation. In *Proceedings of the 3rd International Conference on Web Search and Data Mining*, WSDM '10, pages 91–100, New York, NY, USA, 2010. ACM Press. 18, 103

[3] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 19–26, New York, NY, USA, 2006. ACM Press. 11

[4] R. Andersen, C. Borgs, J. Chayes, J. Hopcraft, V. S. Mirrokni, and S.-H. Teng. Local computation of pagerank contributions. In *Proceedings of the 5th international conference on Algorithms and models for the web-graph*, pages 150–165, San Diego, CA, USA, 2007. Springer-Verlag. 14

[5] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval - The Concepts and Technology Behind Search (Second Edition)*. Pearson Education Ltd., Harlow, England, 2011. 80

[6] L. Baltrunas and X. Amatriain. Towards time-dependant recommen-

dation based on implicit feedback. In *Context-Aware Recommender Systems*, CARS '09, 2009. 5

[7] Z. Bar-Yossef and L.-T. Mashiach. Local approximation of pagerank and reverse pagerank. In *Proceedings of the 17th ACM conference on Information and knowledge management*, number April in CIKM '08, pages 279–288, Napa Valley, California, USA, 2008. ACM Press. 6

[8] Z. Bar-Yossef and L.-T. Mashiach. Local approximation of pagerank and reverse pagerank. In *Proceedings of the 17th ACM conference on Information and knowledge management*, number April in CIKM '08, pages 279–288, Napa Valley, California, USA, 2008. ACM Press. 14

[9] A. Barrat, M. Barthlemy, and A. Vespignani. *Dynamical Processes on Complex Networks*. Cambridge University Press, New York, NY, USA, 2008. 60

[10] A. Bellogín, I. Cantador, and P. Castells. A comparative study of heterogeneous item recommendations in social systems. *Information Sciences*, 221:142–169, Feb. 2013. 81

[11] K. Bharat, A. Broder, M. Henzinger, P. Kumar, and S. Venkatasubramanian. The connectivity server: fast access to linkage information on the web. In *Proceedings of the World Wide Web Conference*, volume 30 of *WWW '98*, pages 469–477, Brisbane, Australia, 4 1998. Elsevier Science Publishers B. V. 14

[12] M. Bianchini, M. Gori, and F. Scarselli. Inside PageRank. *ACM Transactions on Internet Technology*, 5(1):92–128, 2005. 134

[13] P. Boldi, M. Santini, and S. Vigna. Do your worst to make the best : Paradoxical effects in pagerank incremental computations. In *In Proceedings of the third Workshop on Web Graphs (WAW)*, pages 168–180. Springer, 2004. 14

[14] P. Boldi, M. Santini, and S. Vigna. Do Your Worst to Make the Best: Paradoxical Effects in PageRank Incremental Computations. *Algorithms and Models for the Web-Graph*, pages 168–180, 2004. 13

[15] P. Boldi and S. Vigna. Axioms for centrality. *CoRR*, abs/1308.2140, 2013. 12

[16] L. Breiman. Random forests. *Mach. Learn.*, 45(1):5–32, oct 2001. 61

[17] M. Bressan, E. Peserico, U. Padova, and L. Pretto. The power of local information in pagerank. In *WWW Companion*, pages 179–180, Rio de Janeiro, Brazil, 2013. International World Wide Web Conferences Steering Committee. 6, 14

[18] M. Bressan and L. Pretto. Local computation of pagerank: the ranking

side. In *Proceedings of the 20th ACM conference on Information and knowledge management*, CIKM '11, pages 631–640. ACM Press, 2011. 6, 13, 14, 46, 60

[19] J. Burn-Murdoch. Us web statistics. http://www.theguardian.com/news/datablog/2012/jun/22/website-visitor-statistics-nielsen-may-2012-google, May 2012. 1

[20] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso. Comparison of collaborative filtering algorithms. *ACM Transactions on the Web*, 5(1):1–33, Feb. 2011. 18

[21] L. Cao, J. Luo, and T. S. Huang. Annotating photo collections by label propagation according to multiple similarity cues. In *Proceedings of the 16th ACM international conference on Multimedia*, MM '08, pages 121–130, New York, NY, USA, 2008. ACM Press. 16

[22] A. Capocci, A. Baldassarri, V. Servedio, and V. Loreto. Friendship, collaboration and semantics in Flickr: from social interaction to semantic similarity. In *MSM*, pages 8:1–8:4. ACM Press, 2010. 17

[23] C. Castillo, M. El-haddad, J. Pfeffer, and M. Stempeck. Characterizing the Life Cycle of Online News Stories Using Social Media Reactions. In *Proceedings of the 17th ACM Conference on Computer-Supported Cooperative Work and Social Computing*, number February in CSCW '14, Baltimore, USA, 2014. 84, 102, 150

[24] L. D. Catledge and J. E. Pitkow. Characterizing Browsing Strategies in the World-Wide Web. *Computer Networks and ISDN Systems*, 27(6):1065–1073, 1992. 24

[25] O. Celma Herrada. *Music recommendation and discovery in the long tail*. PhD thesis, 2008. 5

[26] M. Cha, F. Benevenuto, Y.-Y. Ahn, and K. P. Gummadi. Delayed information cascades in flickr: Measurement, analysis, and modeling. *Computer Networks*, 56(3):1066 – 1076, 2012. 109

[27] M. Cha, A. Mislove, B. Adams, and K. P. Gummadi. Characterizing social cascades in flickr. *Proceedings of the first workshop on Online social networks - WOSP '08*, page 13, 2008. 109

[28] M. Cha, A. Mislove, and K. P. Gummadi. A measurement-driven analysis of information propagation in the flickr social network. *Proceedings of the 18th international conference on World wide web WWW 09*, page 721, 2009. 109

[29] Y.-Y. Chen, Q. Gan, and T. Suel. Local methods for estimating pager-

ank values. page 381, 2004. 6, 13, 46

[30] M.-F. Chiang, W.-C. Peng, and P. Yu. Exploring latent browsing graph for question answering recommendation. *World Wide Web Internet And Web Information Systems*, pages 1–28, 2011. 12

[31] L. Chiarandini, P. Grabowicz, M. Trevisiol, and A. Jaimes. Leveraging browsing patterns for topic discovery and photostream recommendation. In *International AAAI Conference on Weblogs and Social Media*, 2013. 9, 67

[32] L. Chiarandini, M. Trevisiol, and A. Jaimes. Discovering social photo navigation patterns. In *Proceedings of the IEEE International Conference on Multimedia and Expo*, ICME '12, pages 31–36. IEEE, 2012. 8, 33, 47, 85

[33] F. Chierichetti and R. Kumar. Optimizing Two-Dimensional Search Results Presentation. *Science*, pages 257–266, 2011. 12

[34] J. Cho, H. Garcia-Molina, and L. Page. Efficient crawling through url ordering. In *Proceedings of the World Wide Web Conference*, volume 30 of *WWW '98*, pages 161–172, Brisbane, Australia, 4 1998. Elsevier Science Publishers B. V. 14

[35] O. Chum, J. Philbin, and A. Zisserman. Near duplicate image detection: min-hash and tf-idf weighting. In *BMVC*, volume 3, page 4, 2008. 12

[36] A. S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, page 271, New York, New York, USA, 2007. ACM Press. 19

[37] J. V. Davis and I. S. Dhillon. Large scale analysis of web revisitation patterns. In *12th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, volume 08 of *KDD '06*, pages 116–125, Philadelphia, PA, USA, 2006. ACM Press. 6, 14

[38] M. De Choudhury, H. Sundaram, Y.-R. Lin, A. John, and D. D. Seligmann. Connecting content to community in social media via image content, user tags and user communication. pages 1238–1241, June 2009. 110

[39] G. M. Del Corso, A. Gullí, and F. Romani. Ranking a stream of news. In *Proceedings of the 14th international conference on World Wide Web*, WWW '05, pages 97–106, New York, NY, USA, 2005. ACM Press. 19, 84

[40] W. Eom, S. Lee, W. De Neve, and Y. M. Ro. Improving image tag

recommendation using favorite image context. In *Proceedings of the 18th IEEE International Conference on Image Processing*, ICIP '11, pages 2445 –2448, Sept. 2011. 110

[41] J. Fan, D. Keim, Y. Gao, H. Luo, and Z. Li. Justclick: Personalized image recommendation via exploratory search from large-scale flickr images. *Circuits and Systems for Video Technology*, 19(2):273–288, 2009. 15

[42] F. Figueiredo, F. Benevenuto, and J. Almeida. The Tube over Time : Characterizing Popularity Growth of YouTube Videos. In *Proceedings of the 4th International Conference on Web Search and Data Mining*, WSDM '11, 2011. 13

[43] B. Gao, T.-Y. Liu, Y. Liu, T. Wang, Z.-M. Ma, and H. Li. Page importance computation based on Markov processes. *Information Retrieval*, 14(55):488–514, 2011. 12

[44] B. Geng, L. Yang, C. Xu, X. Hua, and S. Li. The role of attractiveness in web image search. In *Multimedia*, pages 63–72. ACM Press, 2011. 17

[45] J. Gozali, M. Kan, and H. Sundaram. Hidden markov model for event photo stream segmentation. In *International Conference on Multimedia and Expo Workshops*, pages 25–30. IEEE, 2012. 16

[46] A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries*, pages 326–335, 2002. 77

[47] A. Gürsel and S. Sen. Producing timely recommendations from social networks through targeted search. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 805–812. International Foundation for Autonomous Agents and Multiagent Systems, 2009. 110

[48] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In *Proceedings of Very Large Data Bases*, VLDB '04, pages 576–587, Toronto, ON, Canada, 2004. 12

[49] R. Halonen, S. Westman, and P. Oittinen. Naturalness and interestingness of test images for visual quality evaluation. In *SPIE*, volume 7867, page 34, 2011. 17

[50] J. L. Herlocker, J. a. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5–53, Jan. 2004. 17, 18

[51] E. Hoque, O. Hoeber, and M. Gong. Evaluating the Trade-Offs between Diversity and Precision for Web Image Search Using Concept-Based Query Expansion. In *WI-IAT*, volume 3, pages 130–133. IEEE, 2011. 12

[52] J. Hu, H.-J. Zeng, H. Li, C. Niu, and Z. Chen. Demographic prediction based on user's browsing behavior. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 151–160, New York, NY, USA, 2007. ACM Press. 11

[53] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *Proceedings of IEEE International Conference on Data Mining*, ICDM '08, pages 263–272. IEEE Computer Society, 2008. 5

[54] V. Jain and M. Varma. Learning to Re-Rank: Query-Dependent Image Re-Ranking Using Click Data. In *Proceedings of the World Wide Web Conference*, WWW '11, pages 277–286. ACM Press, 2011. 12

[55] Y. Jing. Pagerank for product image search. In *Proceedings of the World Wide Web Conference*, WWW '08, pages 307–315. ACM Press, 2008. 13

[56] Y. Jing and S. Baluja. VisualRank: applying PageRank to large-scale image search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1877–90, Nov. 2008. 12

[57] H. Katti, K. Bin, T. Chua, and M. Kankanhalli. Pre-attentive discrimination of interestingness in images. In *Proceedings of the International Conference of Multimedia and Expo*, pages 1433–1436. IEEE, 2008. 17

[58] J. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999. 12, 46

[59] R. Kumar and A. Tomkins. A characterization of online browsing behavior. In *Proceedings of the World Wide Web Conference*, WWW '10, page 561. ACM Press, 2010. 84, 87

[60] J. G. Lee, P. Antoniadis, and K. Salamatian. Faving reciprocity in content sharing communities: A comparative analysis of flickr and twitter. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*, pages 136–143. IEEE, 2010. 109

[61] R. Lempel and S. Moran. The stochastic approach for link-structure analysis (SALSA) and the TKC effect. *World Wide Web Internet And Web Information Systems*, 33:387–401, 2000. 12

[62] R. Lempel and S. Moran. Salsa : The stochastic approach for link-

structure analysis. *Challenge*, 19(2):131–160, 2001. 46

[63] K. Lerman and L. Jones. Social browsing on flickr. *CoRR*, abs/cs/0612047, 2006. 15

[64] K. Lerman and L. Jones. Social browsing on flickr. *Arxiv preprint cs/0612047*, pages 1–4, 2006. 17, 109

[65] K. Lerman, A. Plangprasopchok, and C. Wong. Personalizing image search results on flickr. *Association for the Advancement of Artificial Intelligence*, pages 65–75, 2007. 17, 135

[66] C. W.-k. Leung, S. C.-f. Chan, and F.-l. Chung. Applying cross-level association rule mining to cold-start recommendations. In *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology - Workshops*, WI-IATW '07, pages 133–136, Washington, DC, USA, 2007. IEEE Computer Society. 5

[67] L. Li, D. Wang, T. Li, D. Knox, and B. Padmanabhan. SCENE: a scalable two-stage personalized news recommendation system. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, SIGIR '11, page 125, New York, New York, USA, 2011. ACM Press. 19

[68] C. Lin, R. Xie, L. Li, Z. Huang, and T. Li. Premise: personalized news recommendation via implicit social experts. In X. wen Chen, G. Lebanon, H. Wang, and M. J. Zaki, editors, *Proceedings of the 21st ACM conference on Information and knowledge management*, CIKM '12, pages 1607–1611. ACM Press, 2012. 19

[69] M. Lipczak, M. Trevisiol, and A. Jaimes. Analyzing favorite behavior in flickr. In *Advances in Multimedia Modeling*, MMM '13, pages 535–545. Springer, 2013. 10, 107

[70] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang. Tag ranking. In *Proceedings of the World Wide Web Conference*, WWW '09, pages 351–360. ACM Press, 2009. 13

[71] J. Liu, P. Dolan, and E. R. Pedersen. Personalized news recommendation based on click behavior. In *Proceedings of the 15th international conference on Intelligent user interfaces*, IUI '10, pages 31–40, New York, NY, USA, 2010. ACM Press. 19

[72] Y. Liu, B. Gao, T.-Y. Liu, Y. Zhang, Z. Ma, S. He, and H. Li. Browserank: letting web users vote for page importance. 31:451–458, 2008. 3, 5, 6, 11, 12, 21, 22, 23, 45, 46, 127, 131, 132, 133, 134, 148

[73] Y. Liu, T.-Y. Liu, B. Gao, Z. Ma, and H. Li. A framework to com-

pute page importance based on user behaviors. *Information Retrieval*, 13(1):22–45, 6 2009. 6, 11, 12, 23, 46

[74] Y. Liu, M. Zhang, and S. Ma. User browsing graph: Structure, evolution and application. In *WSDM (Late Breaking-Results)*. ACM Press, 2009. 11

[75] D. Lu and Q. Li. Personalized search on Flickr based on searcher's preference prediction. In *Proceedings of the World Wide Web Conference*, WWW '11, pages 81–82, 2011. 109

[76] Y. Lv, T. Moon, P. Kolari, Z. Zheng, X. Wang, and Y. Chang. Learning to model relatedness for news recommendation. In *Proceedings of the 20th international conference on World Wide Web*, WWW '11, page 57, New York, New York, USA, 2011. ACM Press. 19

[77] A. McCallum, K. Nigam, and L. H. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. *SIGKDD*, pages 169–178, 2000. 39

[78] R. M. C. McCreadie, C. Macdonald, and I. Ounis. News article ranking: leveraging the wisdom of bloggers. In *Adaptivity, Personalization and Fusion of Heterogeneous Information*, RIAO '10, pages 40–48, Paris, France, France, 2010. Centre de Hautes Etudes Internationales d'Informatique Docmentaire. 84, 102

[79] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa. Discovery and Evaluation of Aggregate Usage Profiles for Web Personalization. *Data Mining and Knowledge Discovery*, 6(1):61–82, 2002. 18

[80] A. L. Montgomery, S. Li, K. Srinivasan, and J. C. Liechty. Modeling online browsing and path analysis using clickstream data. *Marketing Science*, 23(4):579–595, Sept. 2004. 84

[81] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. *World Wide Web Internet And Web Information Systems*, 54(2):1–17, 1998. 6, 12, 46, 59

[82] D. Parra and A. Karatzoglou. Implicit feedback recommendation via implicit-to-explicit ordinal logistic regression mapping. In *Context-Aware Recommender Systems*, number 1 in CARS '11, 2011. 5

[83] D. Parra-Santander and X. Amatriain. Walk the Talk: Analyzing the relation between implicit and explicit feedback for preference elicitation. In *UMAP*, pages 255–268, 2011. 5

[84] J. S. Pedro, P. Street, and S. Sheffield. Ranking and Classifying Attractiveness of Photos in Folksonomies. In *Proceedings of the World Wide Web Conference*, WWW '09, pages 771–780, Madrid, Spain,

2009. ACM Press. 16, 134

[85] C. Prieur, D. Cardon, J.-S. Beuscart, N. Pissard, and P. Pons. The Stength of Weak cooperation : A Case Study on Flickr. *arxiv.org*, 65(8):610–613, 2008. 16, 109, 135

[86] A. M. Rashid, G. Karypis, and J. Riedl. Learning preferences of new users in recommender systems: An information theoretic approach. *SIGKDD Explor. Newsl.*, 10(2):90–100, Dec. 2008. 18

[87] J. Ratkiewicz, A. Flammini, and F. Menczer. Traffic in Social Media I: Paths Through Information Networks. In *Proceedings of the IEEE 2nd International Conference on Social Computing*, pages 452–458. Ieee, Aug. 2010. 13

[88] K. Ren and J. Calic. Freeeye: interactive intuitive interface for large-scale image browsing. In *Proceedings of the 17th ACM international conference on Multimedia*, pages 757–760. ACM, 2009. 16

[89] F. Ricci, L. Rokach, and B. Shapira. Introduction to recommender systems handbook. *Recommender Systems Handbook*, pages 1–35, 2011. 17

[90] A. Said and A. Bellogín. News Recommendation in the Wild : Recommendation Algorithms in the NRS Challenge. 2013. 19

[91] G. Shaw, Y. Xu, and S. Geva. Using association rules to solve the cold-start problem in recommender systems. In M. Zaki, J. Yu, B. Ravindran, and V. Pudi, editors, *Advances in Knowledge Discovery and Data Mining*, volume 6118 of *Lecture Notes in Computer Science*, pages 340–347. Springer Berlin Heidelberg, 2010. 5, 18

[92] J. Sigmund. Over one-third of all internet users visit newspaper websites. `http://www.naa.org/News-and-Media/Press-Center/Archives/2009/Newspaper-websites-attract-more-than-70-million-visitors.aspx`, Aug. 2009. 1

[93] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceedings of the World Wide Web Conference*, WWW'08, pages 327–336, Beijing, China, 2008. ACM Press. 2

[94] A. J. Smola and B. Schölkopf. A tutorial on support vector regression. Technical report, Statistics and Computing, 2003. 61

[95] H. Sobhanam and a. K. Mariappan. Addressing cold start problem in recommender systems using association rules and clustering technique. In *ICCCI*, pages 1–5. Ieee, Jan. 2013. 5, 18

[96] R. Srikant and Y. Yang. Mining web logs to improve website organization. In *Proceedings of the 10th international conference on World Wide Web*, pages 430–437. ACM, 2001. 15

[97] G. Strong, E. Hoque, M. Gong, and O. Hoeber. Organizing and browsing image search results based on conceptual and visual similarities. *Advances in Visual Computing*, pages 481–490, 2010. 16

[98] I. Trajkovski. Pagerank-Like Algorithm for Ranking News Stories and News Portals. *ICT Innovations*, 231:87–96, 2013. 19

[99] M. Trevisiol, L. M. Aiello, R. Schifanella, and A. Jaimes. Cold-start news recommendation with domain-dependent browse graph. In *Proceedings of the ACM Recommender System conference*, RecSys '14, New York, NY, USA, 2014. ACM Press. 83

[100] M. Trevisiol, L. Chiarandini, L. M. Aiello, and A. Jaimes. Image ranking based on user browsing behavior. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 445–454, New York, NY, USA, 2012. ACM Press. 10, 46, 84, 125

[101] M. Tsagkias and R. Blanco. Language intent models for inferring user browsing behavior. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 335–344, New York, NY, USA, 2012. ACM Press. 11

[102] M. Tsagkias and R. Blanco. Language intent models for inferring user browsing behavior. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, page 335, New York, New York, USA, 2012. ACM Press. 19

[103] M. Tsagkias and R. Blanco. Language intent models for inferring user browsing behavior. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 335–344, New York, NY, USA, 2012. ACM Press. 103

[104] M. Tsagkias, M. de Rijke, and W. Weerkamp. Linking online news and social media. In *Proceedings of the 4th International Conference on Web Search and Data Mining*, WSDM '11, pages 565–574, New York, NY, USA, 2011. ACM Press. 84, 102

[105] M. Valafar, R. Rejaie, and W. Willinger. Beyond friendship graphs: a study of user interactions in Flickr. In *SIGCOMM WOSN*, pages 25–30. ACM Press, 2009. 109

[106] R. van Leuken, L. Garcia, X. Olivares, and R. van Zwol. Visual diversification of image search results. In *Proceedings of the World Wide Web Conference*, WWW '09, pages 341–350. ACM Press, 2009. 12

[107] R. van Zwol. Flickr: Who is looking? In *Proceedings of the IEEE International Conference on Web Intelligence*, WI '07, pages 184–190, Washington, DC, USA, 2007. IEEE Computer Society. 111

[108] R. van Zwol and E. Al. Faceted exploration of image search results. In *Proceedings of the World Wide Web Conference*, WWW '10, page 961. ACM Press, 2010. 12

[109] R. van Zwol, A. Rae, and L. Garcia Pueyo. Prediction of favourite photos using social, visual, and textual signals. In *Multimedia*, pages 1015–1018. ACM Press, 2010. 16

[110] R. van Zwol, A. Rae, and L. G. Pueyo. Prediction of favourite photos using social, visual, and textual signals. In *Multimedia*, MM '10, pages 1015–1018. ACM Press, 2010. 109, 120

[111] M. Venkataraman, K. P. Subbalakshmi, and R. Chandramouli. Measuring and quantifying the silent majority on the Internet. *Proceedings of the 35th IEEE Sarnoff Symposium*, (978):1–5, May 2012. 1, 2

[112] W. Wagner. Steven bird, ewan klein and edward loper: Natural language processing with python, analyzing text with the natural language toolkit. *Lang. Resour. Eval.*, 44(4):421–424, Dec. 2010. 99

[113] C. Wang, M. Zhang, L. Ru, and S. Ma. Automatic online news topic ranking using media focus and user attention based on aging theory. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 1033–1042, New York, NY, USA, 2008. ACM Press. 84

[114] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994. 60, 61

[115] J. S. Whissell and C. L. a. Clarke. Improving document clustering using Okapi BM25 feature weighting. *Information Retrieval*, 14(5):466–487, 2011. 76

[116] R. W. White. Investigating behavioral variability in web search. In *Proceedings of the World Wide Web Conference*, WWW '07, pages 21–30, 2007. 11

[117] R. W. White and J. Huang. Assessing the scenic route: measuring the value of search trails in web logs. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 587–594, New York, NY, USA,

2010. ACM Press. 11

[118] O. Wu, W. Hu, and B. Li. Group ranking with application to image retrieval. In *Proceedings of the 19th ACM conference on Information and knowledge management*, CIKM '10, pages 1441–1444. ACM Press, 2010. 12

[119] C. Xian and S. Hyoseop. Extracting Representative Tags for Flickr Users. In *Proceedings of IEEE International Conference on Data Mining*, ICDM '10, pages 312–317, 2010. 110

[120] W. Xing and A. Ghorbani. Weighted PageRank algorithm. In A. A. Ghorbani, editor, *CNSR*, pages 305–314. IEEE, 2004. 133

[121] S. Xu, T. Jin, and F. Lau. A new visual search interface for web browsing. In *Proceedings of the second ACM international conference on web search and data mining*, pages 152–161. ACM, 2009. 16

[122] J. Yang and J. Leskovec. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining - WSDM '11*, page 177, New York, New York, USA, 2011. ACM Press. 13

[123] X. Yang, Y. Guo, and Y. Liu. Bayesian-inference based recommendation in online social networks. In *IEEE INFOCOM*, pages 551–555. Ieee, Apr. 2011. 18

[124] H. Yu, Y. Liu, M. Zhang, L. Ru, and S. Ma. Web spam identification with user browsing graph. In G. Lee, D. Song, C.-Y. Lin, A. Aizawa, K. Kuriyama, M. Yoshioka, and T. Sakai, editors, *Information Retrieval Technology*, volume 5839 of *Lecture Notes in Computer Science*, pages 38–49. Springer Berlin Heidelberg, 2009. 6

[125] J. Yu, X. Jin, J. Han, and J. Luo. Collection-based sparse label propagation and its application on social group suggestion from photos. *ACM Transactions on Intelligent Systems and Technology*, 2(2):1–21, Feb. 2011. 16

[126] E. Zavesky, S.-F. Chang, and C.-C. Yang. Visual islands: intuitive browsing of visual search results. In *Proceedings of the 2008 international conference on Content-based image and video retrieval*, CIVR '08, pages 617–626, New York, NY, USA, 2008. ACM Press. 16

[127] B. Zhou, Y. Liu, M. Zhang, Y. Jin, and S. Ma. Incorporating web browsing activities into anchor texts for web search. *Information Retrieval*, 14(3):290–314, 2010. 12

Appendix A

Categorizations

In this appendix, we provide the complete lists of the categorizations used in this thesis.

A.1. List of Flickr Browsing Dataset Source Categories

Table A.1 shows the URL categories used to categorize URLs that do not belong to Flickr in its browsing dataset.

A.2. List of Flickr Browsing Dataset Page Layouts

Table A.2 lists all page layouts of the Flickr browsing dataset. There is a total of 96 layouts. The italicized parts of the URLs stand for the identifiers: *groupId* for groups, *userId* for users, *photoId* for photos, and *setId* for albums.

Category	Examples of Content
Search	search.yahoo.com, google.com
Social	facebook.com, tumblr.com
Mail	mail.yahoo.com, gmail.com
Aggregator	reddit.com, stumbleupon.com
Blog	blogspot.com, blogger.com
Photo	flickrhivemind.net, compfight.com
Microblog	twitter.com
Forum	discussion forums
News	news.yahoo.com, cnn.com
Shop	ebay.com, amazon.com
Video	youtube.com, vimeo.com
Geo	maps.google.com, maps.yahoo.com
Wiki	wikipedia.org, wikimedia.org
IP	any IP address
Sport	sports.yahoo.com
Autos	autos.yahoo.com

Table A.1: 15 URL source categories in the Flickr dataset.

URL	Label	Description
/	homepage	Homepage
/about	about	About Flickr
/abuse	abuse	Report Abuse
/account	account	Your Account
/activity	activity	Recent Activity: All activity
/analog	analog	Explore Analog
/apps	apps	Your Apps
/bestpractices	bestpractices	Best Practices for Organizations
/cameras	cameras	Camera Finder
/commons	commons	The Commons
/configurator	configurator	Flickr Configurator
/confirm	confirm	Confirmation page
/creativecommons	creativecommons	Creative Commons
/do	domore	Monkey see? Monkey do!
/do/more	domore	Monkey see? Monkey do!
/explore	explore	Explore
/explore/interesting	explore/interesting	Explore interesting photos
/galleries	galleries	Explore Galleries
/gettyimages	gettyimages	Getty
/gift	gift	Flickr gift

continued . . .

URL	Label	Description
/gp	guestpass	Flickr guestpass
/groups	groups	Groups
/groups/*groupId*	groups/groupId	Group page
/groups/*groupId*/admin	groups/groupId/admin	Group administration
/groups/*groupId*/discuss	groups/groupId/discuss	Group discussion
/groups/*groupId*/members	groups/groupId/members	Group members
/groups/*groupId*/pool	groups/groupId/pool	Group photos
/groups/*groupId*/pool/map	groups/groupId/pool/map	Group photo map
/groups/*groupId*/pool/tags	groups/groupId/pool/tags	Group tags
/groups/*groupId*/pool/with	groups/groupId/pool/with	People appearing in group's photos
/groups/*groupId*/rules	groups/groupId/rules	Group rules
/groups create.gne	groups/action	Create group
/groups invite.gne	groups/action	Invite to group
/groups join.gne	groups/action	Join group
/groups leave.gne	groups/action	Leave group
/guidelines	guidelines	Flickr Community Guidelines
/help	help	Help
/iconbuilder	action iconbuilder	The Icon Builder
/import	import	Find your friends
/import/people	import/people	Find your friends

continued . . .

URL	Label	Description
/invite	invite	Invite your friends
/logout.gne	logout	Log out
/logout ok.gne	logout	Log out
/mail	mail	Flickr Mail: Your Inbox
/mail/contact notifications	mail cnt notify	Flickr Mail: Contact Notifications
/mail/reply	mail reply	Flickr Mail: Reply
/mail/sent	mail sent	Flickr Mail: Your Sent Mail
/mail/write	mail write	Flickr Mail: Compose a Message
/map	map	Explore Anyones' photos on a Map
/nearby	nearby	Everyone's photos taken near you
/partners/getty	partners/getty	Getty
/photo delete.gne	photo gne	Delete photo
/photo edit.gne	photo gne	Edit photo
/photos	photos	Explore
/photos/friends	photos/friends	From the people you know
/photos/organize	photos/organize	Organize your photos
/photos/tags	photos/user/tags	Popular tags on Flickr
/photos/upload	photos/upload	Upload a photo
/photos/upload/basic	photos/upload	Upload a photo
/photos/*userId*	photos/user	Display all user photos

continued . . .

URL	Label	Description
/photos/*userId*/alltags	photos/user/tags	Display all user tags
/photos/*userId*/archives	photos/user/archives	Display all user photos in cronological order
/photos/*userId*/collections	photos/user/collections	View user albums
/photos/*userId*/favorites	photos/user/usersfavs	Display all user favorites
/photos/*userId*/galleries	photos/user/collections	View user albums
/photos/*userId*/map	photos/user/map	Explore user photos on a map
/photos/*userId*/page	photos/user	Display all user photos
/photos/*userId*/people	photos/user/people	People featured in user photos
/photos/*userId*/popular	photos/user/popular	Popular user photos
/photos/*userId*/sets	photos/user/sets	View user albums
/photos/*userId*/sets/*setId*	photos/user/sets	Display all album photos
/photos/*userId*/show	photos/user/photoId	Display single photo
/photos/*userId*/stats	photos/user/stats	User statistics
/photos/*userId*/tags	photos/user/tags	User tags
/photos/*userId*/upload	photos/user/upload	Upload a photo
/photos/*userId*/with	photos/user/with	People appearing in user's photos
/photos/*userId*/*photoId*	photos/user/photoId	Display single photo
/photos/*userId*/*photoId*/favorites	photos/user/photoId/photosfavs	People who favorited the photo
/photos/*userId*/*photoId*/in/contacts	photos/user/photoId/contacts	Browse contacts photos
/photos/*userId*/*photoId*/in/faves-*userId*	photos/user/photoId/faves	Browse user favorites

continued . . .

URL	Label	Description
/photos/*userId*/*photoId*/in/photostream	photos/user/photoId	Browse user photos
/photos/*userId*/*photoId*/in/pool-*groupId*	photos/user/photoId/pool	Browse group photos
/photos/*userId*/*photoId*/in/set-*setId*	photos/user/photoId/set	Browse user album
/photos/*userId*/*photoId*/meta	photos/user/photoId/meta	Photo metadata
/photos/*userId*/*photoId*/sizes	photos/user/photoId/sizes	Photo in different resolutions
/photosets deletecomment.gne	photosets/deletecomment	Delete comment
/photosets editcomment.gne	photosets/editcomment	Edit comment
/photosof	photosof	People you follow
/places	places	Explore places
/profile delete.gne	action profile delete	Delete profile
/search	search	Search photos
/search/advanced	search advanced	Search photos
/search/forum	search forum	Search forum
/search/groups	search groups	Search groups
/search/people	search people	Search people
/search/show	search show	Search photos
/services/api	services/api	Flick API
/services/apps	services/apps	Flick Apps
/services/auth	services/auth	Authentication
/services/developer	services/developer	The Flickr Developer Guide

continued . . .

URL	Label	Description
/services/feeds	services/feeds	Flickr photo feed
/services/oauth	services/oauth	O-auth authentication
/services/partners	services/partners	Flick parterns
/signin	signin	Sign in
/signup	signup	Sign up
/tools	tools	Tools to upload and share photos
/tour	tour	Flickr tour
/upgrade	upgrade	Upgrade account
/welcome	welcome	Welcome to Flickr

Table A.2: List of page layouts in Flickr. The table shows the URL inside Flickr and the description of the layout.

Appendix B

Case Study Survey

In this appendix, we provide the surveys that we used for the evaluation of the experiments conducted in Chapter 6.

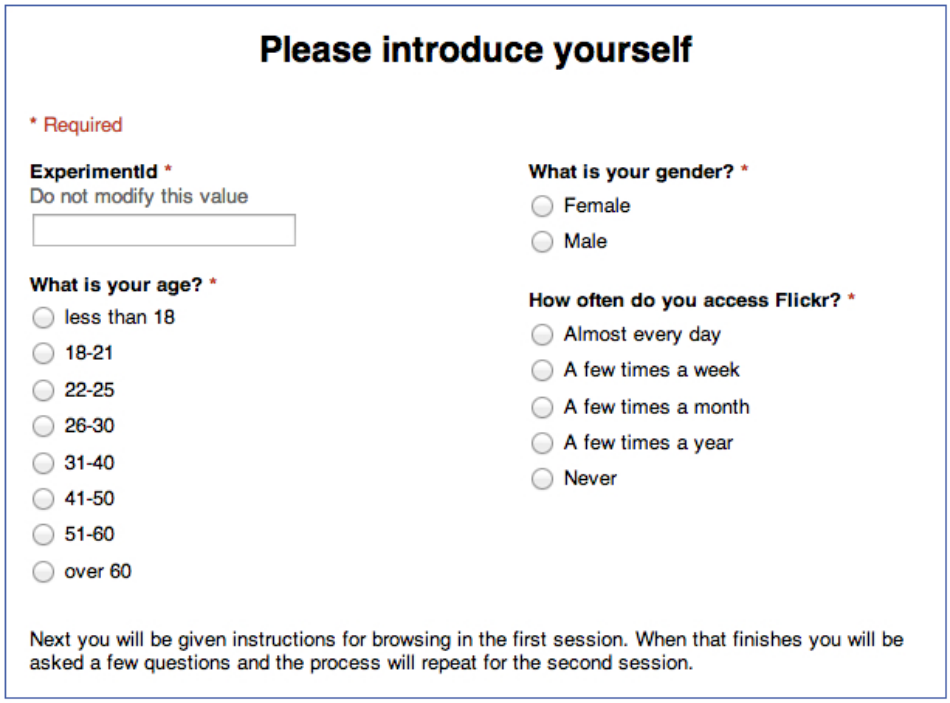
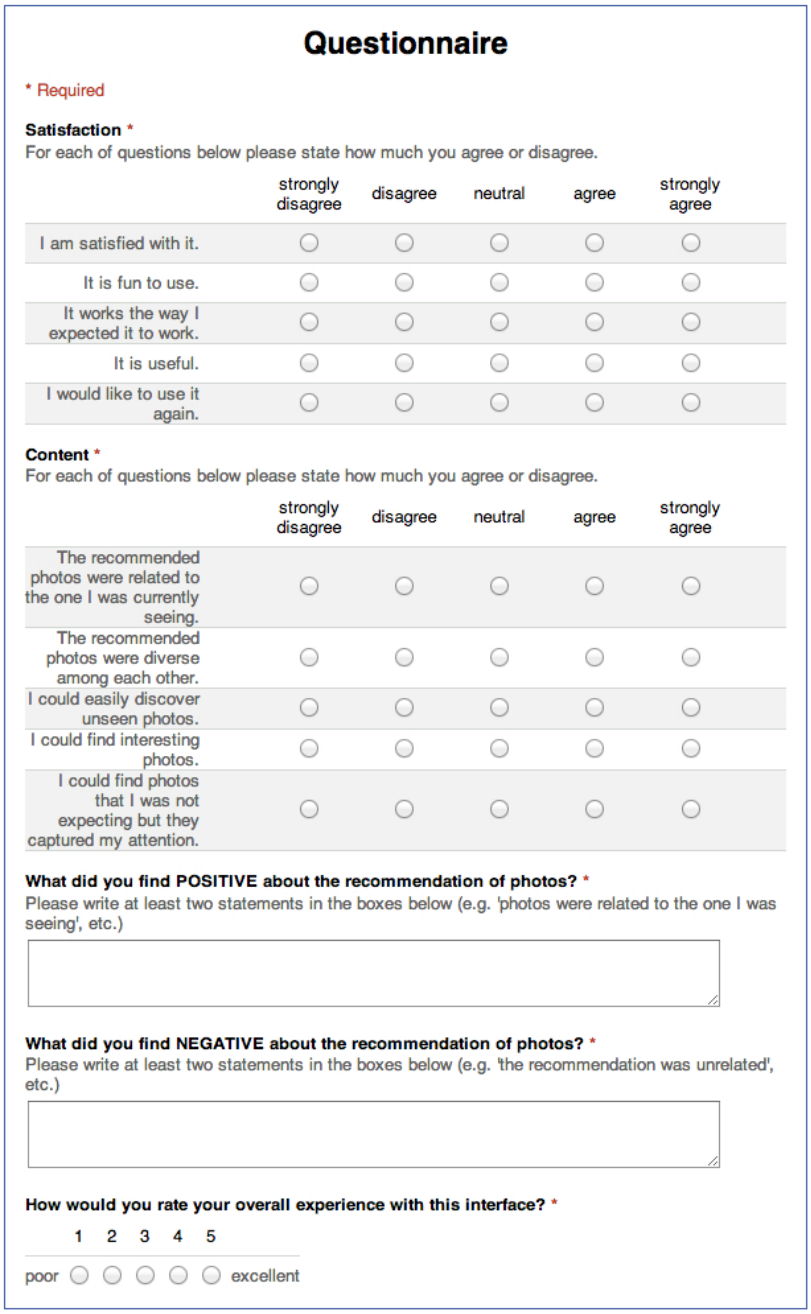


Figure B.1: General information about the user, asked at the beginning of the survey.



Questionnaire

* Required

Satisfaction *

For each of questions below please state how much you agree or disagree.

	strongly disagree	disagree	neutral	agree	strongly agree
I am satisfied with it.	○	○	○	○	○
It is fun to use.	○	○	○	○	○
It works the way I expected it to work.	○	○	○	○	○
It is useful.	○	○	○	○	○
I would like to use it again.	○	○	○	○	○

Content *

For each of questions below please state how much you agree or disagree.

	strongly disagree	disagree	neutral	agree	strongly agree
The recommended photos were related to the one I was currently seeing.	○	○	○	○	○
The recommended photos were diverse among each other.	○	○	○	○	○
I could easily discover unseen photos.	○	○	○	○	○
I could find interesting photos.	○	○	○	○	○
I could find photos that I was not expecting but they captured my attention.	○	○	○	○	○

What did you find POSITIVE about the recommendation of photos? *

Please write at least two statements in the boxes below (e.g. 'photos were related to the one I was seeing', etc.)

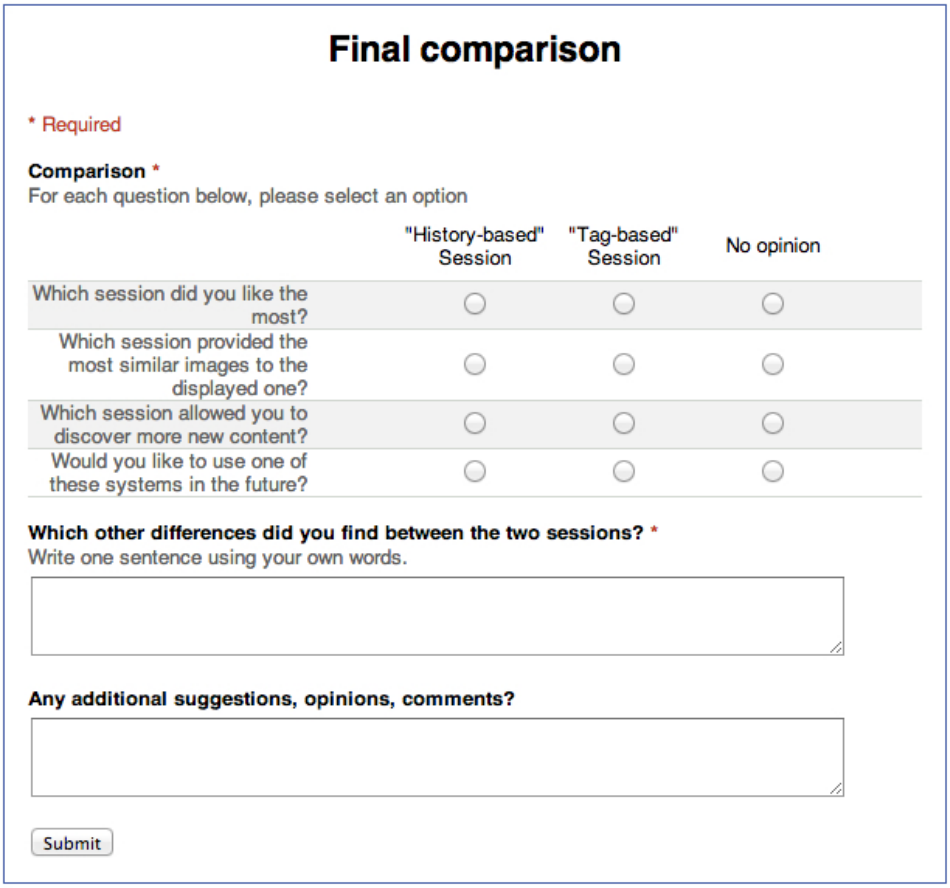
What did you find NEGATIVE about the recommendation of photos? *

Please write at least two statements in the boxes below (e.g. 'the recommendation was unrelated', etc.)

How would you rate your overall experience with this interface? *

	1	2	3	4	5	
poor	○	○	○	○	○	excellent

Figure B.2: Questions asked after the user experience of both the recommender systems.



Final comparison

* Required

Comparison *
For each question below, please select an option

	"History-based" Session	"Tag-based" Session	No opinion
Which session did you like the most?	○	○	○
Which session provided the most similar images to the displayed one?	○	○	○
Which session allowed you to discover more new content?	○	○	○
Would you like to use one of these systems in the future?	○	○	○

Which other differences did you find between the two sessions? *
Write one sentence using your own words.

Any additional suggestions, opinions, comments?

Submit

Figure B.3: Final set of questions asked at the end of the survey.