

PageFetch 2: Gamification the Sequel

Leif Azzopardi, Martin Bevc, Andrew Gardner, David Maxwell and Abdullah Razzouk
School of Computing Science
University of Glasgow
United Kingdom

ABSTRACT

This demo poster presents *PageFetch 2*, the sequel of *PageFetch*, *Fu-Finder* and *PageHunt*. The game is simple. A player is shown a Webpage. The player must then enter a query that they believe will retrieve the displayed page. The shorter the query and the higher the page appears in the returned rankings, the more points the player receives. Players have three minutes to find as many of the displayed pages as possible. What makes this iteration different is the inclusion of several additional *gamification* features other than just points and an updated scoring system. We include leaderboards, badges and an avatar component - all of which are designed to help and motivate players. The framework is also publicly available for download, and can be customised to create new search games - or even to evaluate different search engines and document collections. This demo is a preliminary report of the progress we have made to date. At the workshop, we hope to solicit feedback on how we can improve the game and its gameplay.

Categories and Subject Descriptors

H.5 [Information Interfaces and Presentation]: User Interfaces—*Interaction styles*; H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*Information Retrieval*

General Terms

Design, Human Factors

Keywords

Evaluation, Search Games, PageHunt, Fu-Finder, PageFetch

1. INTRODUCTION

Typically, the primary focus of evaluation is to determine the quality of a retrieval system. However, the performance

of users and their ability to use such systems is also particularly important. The abilities of different user groups may also vary significantly. For example, children often face many difficulties when searching, and also find formal experimental settings boring, intimidating and tedious [9]. However, some experiments lend themselves to be *gamified*. Rather than undertaking an experiment, players are invited to participate by playing a game. The experiment or task at hand is essentially turned into an engaging, fun and challenging experience. The major challenge here is to find a means in which the ‘boring’ experiment or ‘tedious’ task can be turned into a game. If such a goal can be successfully attained, these games can provide us with a controlled setting (much like that of a traditional, ‘boring’ experiment) in which large volumes of useful and reliable data can be generated, all at a very low cost [6].

A number of experiments have already attempted to gamify the evaluation process (see [2, 3, 4, 5, 7, 8]). These cited examples either examine how well users perform, how well a system performs, or seek to obtain relevance judgements. In this work, we extend the game called *PageFetch* [1, 9], which succeeds from a series of related search games, including *PageHunt* [7] and *Fu-Finder* [8].

In each of the aforementioned games, players are shown a Webpage and must attempt to enter a query that will retrieve that page. However, the focus of each game and certain features vary. We shall explain these variations in the next section, before describing *PageFetch 2*, a new version of PageFetch. This new version adds a number of additional features to create a more gamified experience, as well as including a number of settings that provide customisation to the underlying experiment and overall user experience.

2. PAGE FINDING GAMES

The most similar games related to PageFetch are PageHunt [7] and Fu-Finder [8] (see Figure 1 for screenshots of each). The former was developed to identify Webpages that people found difficult to find using the *Bing* search engine, while the latter was developed to compare different search engines, comparing their results. PageFetch was developed to study the querying behaviours of children, and to assess the difference in searching abilities between children and adults [1].

As previously mentioned, players in each of these games were presented with a Webpage. Players then entered a query which they believed would retrieve the page shown. Players then had three minutes to attempt to find as many pages as possible. Some of the differences and similarities

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GamifIR '14, April 13, 2014, Amsterdam, Netherlands
Copyright 2014 ACM ...\$15.00.

between these games are highlighted below.

1. PageHunt and Fu-Finder used a similar set of pages, a mixture of hard and easy to find pages given Bing's search engine at the time. In contrast, the original PageFetch used pages from six different categories: *cartoons*, *music*, *sports*, *politics*, *news* and *films*. This provided a range of genres that appealed to both children and adults¹. Thus, players were given a choice of category from which pages were presented, rather than the uncategorised pages presented in the other two games. The choice of categories provided more focused gameplay, and let users pick a genre that was of interest to them.
2. Unlike PageHunt, PageFetch assigned points according to how high in the returned rankings for a given query the shown page was. The higher the rank, the more points the user received.
3. In Fu-Finder, the query issued by the user was sent to three search engines (Bing, *Google* and *Yahoo!*). At the time, all three had publicly available Web search APIs, and were all independent search services. The game therefore attempted to identify queries where one search engine outperformed the other two.
4. Finally, PageFetch was designed to encourage players to focus on entering successful queries. This was achieved by providing a 'no pass' bonus which increased progressively when the player successfully found a query returning the given page in the ranked results.

3. PAGEFETCH 2

PageFetch 2 follows the same gameplay as the previous version and related search games. In PageFetch 2, we have chosen eight categories: *actors*, *business people*, *kids TV*, *films*, *space objects*, *universities*, *musical artists* and *games* (see Figure 2). The categories provide a variety of different domains, and appeal to different demographics. We also anticipate that the chosen categories will be better suited to gamers of different ages, and thus represent different levels of difficulty. A teenager for example may not know much of the world of business, but may be more likely to know more about games.

To populate these categories, we manually recorded 30 trending queries per category (using *Google Trends*). These queries were then submitted to Bing through their Search API to retrieve the top 50 results. One Webpage was then randomly selected from the top 10 results, the following 10 results, and so on. The rationale behind this process was that the lower the page in the results list, the harder it would be to find. This would mean the player would have to go beyond the popular (and short) query, and submit additional terms in order to find the page. However, it is still possible to manually select pages or populate the game pages in other ways. Another game we have developed called *Retrieve Me If You Can* demonstrates this - we used pages from our university to create a site search version of the game.

¹Pages were selected using trending search terms for under-12s, under-16s and adults from 2011.

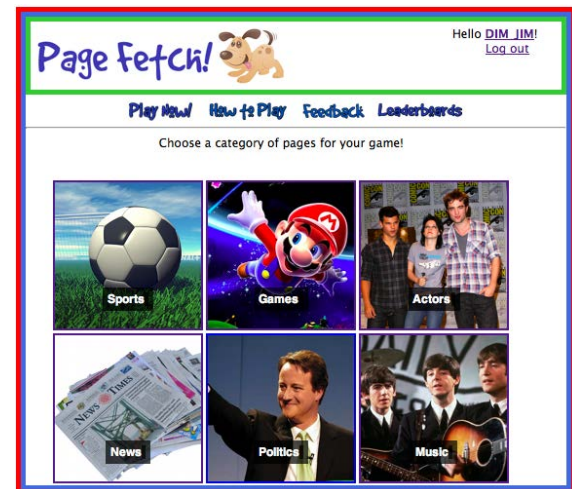
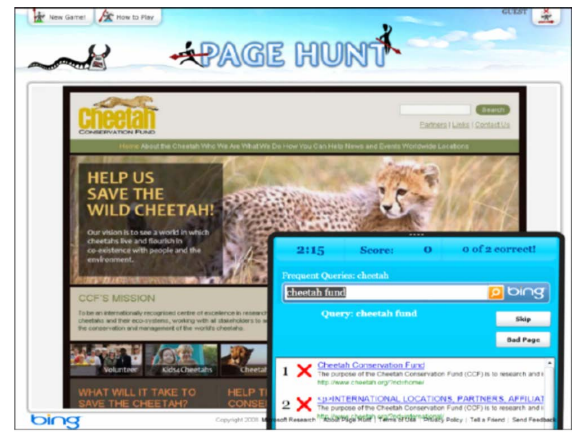


Figure 1: Screenshots of PageHunt (top), Fu-Finder (middle), and PageFetch (bottom) in various stages of the game.

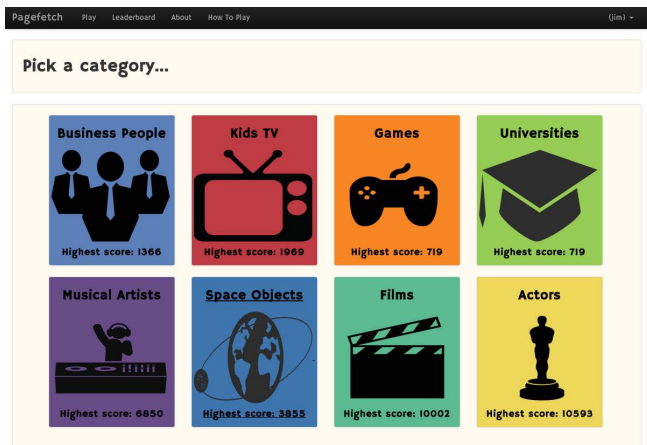


Figure 2: Categories page in PageFetch 2. Players first pick a category that they want to try.

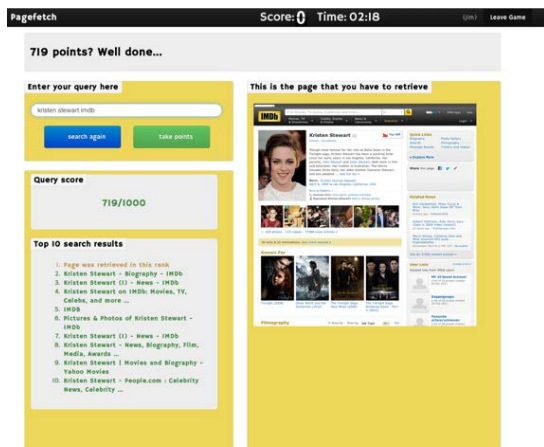


Figure 3: The gameplay page in PageFetch 2.

Once a player selects a category, they are given three minutes to find as many pages as they can. The current interface for this is shown in Figure 3. Players are shown a screenshot of the page they need to find on the right. Queries can be issued on the left, and see the results underneath. Below the search box, users also see the number of points obtained for the issued query. There are also two buttons for the player to issue their query, or to take the points offered - and thus move to the next page. The top bar of the interface provides information about the time remaining, and the user's overall score. Underneath the top bar is an area for messages and feedback from the game's avatar.

PageFetch 2 provides a more game-like experience by including the following features.

- **Points:** The scoring system in PageFetch 2 has been changed so that it rewards players for posing short queries that yield the page high up the returned rankings. We shall discuss this in more detail below.
- **Leaderboards:** While these were part of the original version of PageFetch, we have provided leaderboards in the new version to show how different age groups, genders, cities and countries perform. At a localised level, we can also include different schools, colleges and

universities. These are included to help foster competition, and to help motivate groups to play rather than just as individuals.

- **Badges:** Previous versions of the game have not rewarded players in terms of achievements to recognise their progress, skill or accomplishments. We have included a number of badges to encourage the players to play as many games, to play all the available categories, and to help them become better searchers.

PageFetch 2 has been completely re-implemented to make it more extensible and customisable, and now includes a number of new features as explained above. The developed framework is called *ifind*, and is hosted on *GitHub*². The core of the framework has been written in *Python*, with web applications built with the *Django* Web framework. The *ifind* framework includes several wrappers for a variety of search engines, and is easy to extend for others. There are also several components to handle game mechanics, leaderboards, awards and achievements.

3.1 Points System

In previous games, the points system was either based on the number of pages found (see [7]), or the number of pages and the rank at which those pages were retrieved (see [1, 8]).

For PageFetch 2, we have opted to experiment with the points system to try and encourage players to be as savvy as possible when querying. Players are encouraged to pose the shortest possible query to retrieve the requested page. The idea here is to combat the obvious way to retrieve the page, by posing a very long query using a sentence, title or heading from the shown page. Players are therefore faced with a trade-off - a long query which is likely to retrieve the page will yield fewer points than a shorter query with a smaller chance of returning the page but would return a greater number of points.

We feel that this scheme encourages more realistic querying behaviour as users of web search engines tend to pose short queries. This variation to the game is also designed to encourage users to pose multiple queries for the page through query reformulation. However, it is likely that when the game is deployed we will have to balance the trade-off between the length of queries and the difficulty of the page. This is to ensure that players do not skip pages they perceive as difficult, and instead attempt to find all pages.

3.2 Category Ordering and Rotations

It was found with the original PageFetch that players focused on the top three categories presented, and seldom played the bottom three categories. In PageFetch 2, we have addressed this by introducing a class to handle different ways to rotate categories. For a given player, either: (i) a rotation is selected randomly, and assigned to that player so that each time they login, they are presented with the same rotation; or (ii) each time they view the category grid, they are presented with a random rotation. By doing this, we hope to reduce the effect of static ordering and obtain a relatively equal number of games across all categories. We however do anticipate a topic bias due to the demographics of given users, and will need to consider how to handle this when examining results.

²The *ifind* framework and associated applications are available at <https://github.com/leifos/ifind>.

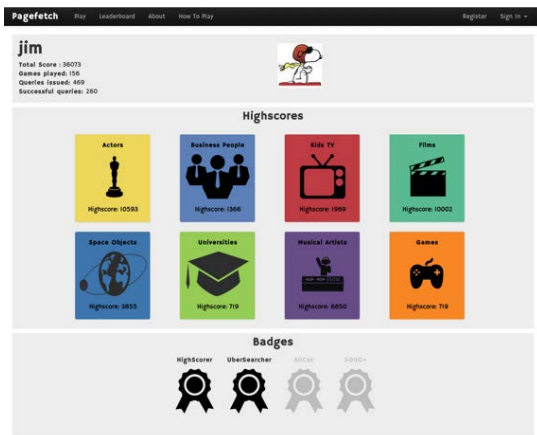


Figure 4: Profile Page in PageFetch 2.

3.3 Learning: Avatars and Rewards

Two features in this version are designed to encourage learning: (i) the avatar; and (ii) badges. A set of classes have been created which encapsulate the behaviour of a helpful avatar, which will aid the player through the game by providing encouraging messages, search tips and other forms of advice. The avatar is privy to the information about the context of an instance of the game.

As an example, the avatar is made aware of a player who has issued three queries in a row, all of which have failed to retrieve the desired page. The avatar can use this information to formulate encouraging messages like “keep going” and “try again”, or even provide tips like “try adding the site name to your query” and “use quotes to focus your query”. Once we have obtained a set of good queries for the page, the avatar will also be able to suggest keywords, or even suggest if the player is on track.

In addition to avatars, badges provide goals for the player to aspire to (see Figure 4). They are designed to encourage players to play multiple games, to play various categories, and to encourage them to learn how to search more effectively. To provide such functionality, an achievement class has been derived. Within this class, the conditions under which the badge is awarded are defined. Currently, we have badges for playing 10 games, and a badge for playing at least one game in each category to provide a sense of accomplishment. We also provide a series of badges for scoring more than 3,000 points in a single game, and for finding three pages in a row - both of which are designed to encourage perseverance and patience.

3.4 Demonstrator

We are currently in the final phases of development and are currently testing the game. Currently hosted at <http://pagefetch.pythonanywhere.com>, the game will eventually be hosted at <http://www.pagefetch.org>. The application is being regularly updated, so there may be periods where the game will be offline. At the workshop, we will be giving demonstrations of these games where we will be soliciting feedback on how to improve them.

4. SUMMARY AND FUTURE WORK

In this demo paper, we have described PageFetch 2, a

search game designed for children and adults. The game follows on from PageFetch and other related games, but has been created within a framework to allow students, researchers and others to create similar games. These can be used to evaluate the quality of competing search engines, or evaluate the abilities of various user populations.

In future work, we plan to conduct a large scale study using PageFetch 2 to examine the performance of high school students, college and university students, and the general population. Specifically, we wish to see how their performance on the game correlates with their demographics, and whether repeated play can improve their search performance. By including different conditions - such as a helpful avatar that provides search tips when they start, versus no search tips - we will be able to see whether such advice can improve the performance of players.

Acknowledgements: This research was partly funded by the EPSRC Project, *Models and Measures of Findability* (EP/K000330/1).

5. REFERENCES

- [1] L. Azzopardi, J. Purvis, and R. Glassey. Pagefetch: A retrieval game for children (and adults). In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 1010–1010, 2012.
- [2] M. Efron. A game with many purposes: Improving information retrieval through pleasurable competition. In *CIS 2011 Workshop: Collaborative Information Seeking. The Annual Meeting of the American Society for Information Science and Technology*, 2011.
- [3] G. Kazai, N. Milic-Frayling, and J. Costello. Towards Methods for the Collective Gathering and Quality Control of Relevance Assessments. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'09, pages 452–459, Boston, MA, USA, July 2009. ACM.
- [4] K. Halttunen and E. Sormunen. Learning information retrieval through an educational game. is gaming sufficient for learning? *Education for Information*, 18(4):289–311, 2000.
- [5] C. G. Harris and P. Srinivasan. Applying human computation mechanisms to information retrieval. *Proceedings of the American Society for Information Science and Technology*, 49(1):1–10, 2012.
- [6] L. von Ahn and L. Dabbish. Designing Games With a Purpose. *Communications of the ACM*, 51:58–67, August 2008.
- [7] H. Ma, R. Chandrasekar, C. Quirk, and A. Gupta. Improving search engines using human computation games. In *Proceeding of the 18th CIKM*, pages 275–284, 2009.
- [8] C. O’Neil, J. Purvis, and L. Azzopardi. Fu-finder: a game for studying querying behaviours. In *Proceedings of the 20th ACM CIKM*, pages 2561–2564, 2011.
- [9] J. Purvis and L. Azzopardi. A preliminary study using pagefetch to examine the searching ability of children and adults. In *Proceedings of the 4th Information Interaction in Context Symposium*, IIX '12, pages 262–265, 2012.