Abstract Web spam potentially causes three deleterious effects: unnecessary work for crawlers and search engines; diversion of traffic away from legitimate businesses; and annoyance to search engine users through poorer results.

Past research on web spam has focused on spamming techniques, spam suppression techniques, and methods for classifying web content as spam or non-spam. Here we focus on the deterioration of search result quality caused by the presence of spam in a country-scale web. We present a framework for measuring the degradation in quality of search results caused by the presence of web spam. We index the 80 million page UK2006 web spam collection on one machine. We trial the proposed framework in an experiment with the UK2006 collection and demonstrate that simple removal of spam pages from result sets can increase result quality. We conclude that the framework is a reasonable vehicle for research in this area and outline changes necessary for planned future experiments.

Keywords Web Information Retrieval, Web Spam, Adversarial Information Retrieval

1 The web spam problem

Web search engines are the first port of call for many users of the World Wide Web. This creates a strong commercial pressure to achieve a high web search rank. Many site operators strive to improve the layout and content of their sites using Search Engine Optimisation (SEO). However, some operators attempt to fool the ranking algorithms used by web search engines, using techniques commonly referred to as black hat SEO or Web spam. Web spam is a problem because it negatively affects the quality of the result list.

Proceedings of the 12th Australasian Document Computing Symposium, Melbourne, Australia, December 10, 2007. Copyright for this article remains with the authors.
pages in the open directory project\textsuperscript{1} as good. These combine to give a total of 10,662 judgements, covering most of the hosts in the collection. Each label is one of \{“normal”, “borderline”, “spam”, “can not classify”\}, using the guidelines\textsuperscript{2} provided to the judges. Following the approach in [3], for our experiments we consider only hosts marked by two human judges or from controlled domains. Of these labels, 5,549 of them are “normal” and 1,924 of them are “spam”.

2.1 Indexing

We indexed the 2 terabytes of the UK2006 collection on a low cost machine with 2 Intel P4 3.0GHz CPUs, 3GB of RAM, and just over 1TB of disk space. As the collection is compressed, this disk space is sufficient. The total system cost was approximately $2,000 AUD.

PADRE [6] was used for indexing and query processing. It supports searching of dynamically defined meta-collections, each comprising indexes of up to 16 primary collections. The experiments reported here used a meta-collection of four primary collections, comprising the whole UK2006 collection. This reduces the risk of exhausting disk space during indexing. Meta-collections accurately simulate the effect of indexing all the data as a single index, except that extra effort is required to correctly index links which cross from one primary collection to another. The total time to decompress and index the entire collection was 166.3 hours, resulting in a 129.5 GB index.

2.2 Query processing

To emulate commercial search, we used Document-At-A-Time (DAAT) \cite{2, 8} query processing. DAAT allows early termination of postings scans because document numbers are assigned in order of a descending query-independent static score. Unfortunately, due to time and hardware constraints, our document numbers were only assigned in collection order. Query response time was very reasonable. The time to generate and present 100 results for 100 queries to another machine on the network via the web interface was only 25 seconds, provided the search engine was in a “warm” state.

The quality of the search results produced is subjectively poor. For our initial experiment, we wanted to determine whether a baseline ranking can be improved simply by removing known spam items. For this it is not necessary that the baseline be of the highest quality.

The interested reader is invited to examine our baseline retrieval engine\textsuperscript{3}. Further information about and access to the experiment can be found at \url{uk.wirrapoi.com}.

3 Does web spam affect result quality?

An easy treatment of web spam is to simply remove it from the result list. This is attractive because it enables the easy combination of spam detection techniques and because indexes do not have to be rebuilt. However, it does little to combat anchor text or link spam whose target is not a spam page. We test whether simply removing spam from result pages improves quality.

3.1 The presence of spam

Clearly, if spam pages never appear in our results, there will be nothing to remove. Consequently, we checked to see how much spam is present in typical result pages. For this, we obtained queries from the dogpile search spy\textsuperscript{4}, a tool for viewing live searches on the dogpile.com search engine. Since spam is denser around popular queries \cite{4}, we selected every query that appeared twice or more in a 72 hour period. After filtering these queries to remove searches that included domain names not present in our collection, we had 328 unique queries. The top ten results were produced for each query, and the number of spam labelled hosts was counted (Figure 1). Clearly, our ranking has been influenced by this spam, as an average 32\% of these results are labelled as spam, compared with 17\% in the overall collection. This demonstrates that web spam is overrepresented in typical result pages.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{spam_in_top_tens.png}
\caption{The amount of spam present in the top ten results for 328 queries from the dogpile search spy. Queries are sorted by their popularity, with the most popular being the far left.}
\end{figure}

3.2 Experimental setup

Volunteer subjects submitted queries of their own using our two-panel evaluation interface (see \cite{10}, shown in Figure 5). Two result pages are presented side by side, and users are invited to judge one list as being better than the other, or “no difference” between the lists. We informed users they were accessing a UK search service and suggested that, if they had difficulty thinking of queries, to imagine that they were about to travel to the UK. They were presented with two sets of unlabeled

\textsuperscript{1}\url{http://www.dmoz.org/}
\textsuperscript{2}\url{http://www.yr-bcn.es/webspam/datasets/uk2006-info/}
\textsuperscript{3}\url{http://uk.wirrapoi.com/padre-sw.cgi?collection=ukk&query=spam}
\textsuperscript{4}\url{http://www.dogpile.com/info.dogpl/searchspy/}
search results, *standard* and *filtered*. Both derive from a single search processed as described above, which returns up to 100 results. *Standard* comprises the first 10 of these results and *filtered* comprises the first ten after pages from previously labelled\(^5\) spam sites were removed. The left-right order of presentation of *standard* and *filtered* was randomized to avoid bias.

### 3.3 Results

245 preference judgements were collected from 31 users over a period of two days. These judgements covered 239 unique queries. Of these judgements, 78 were votes for the *filtered* result set, 36 were votes for the *standard* result set, and 131 were explicit votes for neither set. In a few cases, the result sets were identical because there was no labelled spam present in the top ten *standard* results. Discarding judgements on identical sets, we get 75 votes for *filtered*, 83 votes of no difference, and 35 votes for *standard*. Overall preferences for each user were also computed. This was done by scoring a vote for *filtered* as +1, no difference as 0, and *standard* as −1, then summing these scores. A user’s preference will be *standard*, no difference, or *filtered* if their sum is less than, equal to, or greater than zero respectively. Under this scheme, 19 users preferred *filtered*, 7 had no preference, and 5 preferred *standard*. Results are presented graphically in Figure 2, while the total number of judgements and judgement sum for each user can be seen in Figure 3.

![Figure 2: Overall totals of judgements. The white bars show total judgements overall, and the black bars show one judgement average preference for a user.](image)

We also counted the number of spam results present in each submitted query. 187 (88%) of all queries had some spam present, with a total of 613 labelled spam pages being presented to users in the *standard* result set (26% of all result pages presented in that panel). We visualise the distribution of user judgements with respect to the amount of spam present in the result set in Figure 4. There appears to be no correlation between the number of judgements made by a user, and judgement preference (Figure 3).

![Figure 3: Judgement totals for individual users. The black lines show the total number of judgements, while the grey lines show the sum of that user’s judgements (plus one for each *filtered* vote, minus one for each *standard* vote.).](image)

![Figure 4: The distribution of judgements for varying amounts of spam in the top ten results. It is interesting that a stronger preference for the *filtered* set does not develop as more spam appears in the results.](image)

### 3.4 Discussion

Ignoring the no difference votes, there is a strongly significant difference between the total votes for *filtered* and *standard* (Pearson’s chi-square test, \(p < 0.0001\). However, *filtered* cannot be said to be strictly better than *standard* as the total *filtered* votes are not greater than the total *standard* votes plus the no difference votes.

Examining Figure 4 there appears to be no correlation between the amount of spam removed in the *filtered* set and the judgement that users make (other than a preference for no difference when no spam is present). It is not yet clear why this is, as intuitively more spam removed would equate to higher quality results.

Anecdotally, users observed that many search results which are not labelled as spam nonetheless did not deserve to be ranked as highly as they were. This may be due to incompleteness of the labelling or deficiencies in the search process.

\(^5\)using the data supplied with the UK2006 collection
Queens University - Two-panel search tool

Figure 5: A screen shot of our two panel judgement interface. Two result pages are presented, and the user is invited to judge the left as better, the right as better, or both the same. The left and right order of the panels are randomised each query.

in our ranking algorithm. It also may be due to non-spam pages benefiting from artificially inflated quantities of links and anchortext. Future work is planned to investigate techniques for nullifying this sort of “optimisation”.

In future work using this framework, we need to think carefully about what constitutes the ideal baseline. We want a ranking which is as high quality as possible, without employing any techniques for counter-optimisation and spam. Because score components such as link counts, PageRank scores, and anchortext scores may change dramatically when counter-optimisation methods are applied, it is clear that we will need separate indexes to support the baseline and the spam-reduced version.

The queries made up by our volunteers are unlikely to be representative of the real work load of a UK search engine. For greater realism, we will recruit volunteers in the UK, or obtain lists of actual UK queries.

4 Conclusion and future work

With basic hardware, we successfully indexed the 2 terabyte, 80 million page UK2006 collection and implemented a UK search engine with sufficiently good result quality and response time to support an initial experiment in spam rejection.

Our evaluation method was sufficiently sensitive to detect differences between baseline and filtered rankings. We showed that spam does affect the quality of results for a large number of queries.

In future work, we plan to implement a better baseline and to compare it with a range of approaches to spam nullification.

Acknowledgements

Many thanks to Paul Thomas for providing his two panel interface code and helpful advice. Also thanks to the many volunteers who supplied searches and judgements.

References