Optimization By Proxy

Alexandros Marinos
Digital Ecosystems Group
Department of Computing
University of Surrey
+44 (0)1483 686058
a.marinos@surrey.ac.uk

Paul Krause
Digital Ecosystems Group
Department of Computing
University of Surrey
+44 (0) 1483 689861
p.krause@surrey.ac.uk

ABSTRACT
In this paper, we describe the pattern of optimization by proxy, which can be found in most large-scale algorithmic systems. The pattern occurs when a computationally limited optimizer uses a proxy property as a shortcut indicator for the presence of a hard-to-measure target quality. However, when intelligent actors with different motivations are involved, the existence of the algorithm reifies the proxy into a separate attribute to be manipulated with the goal of altering the algorithm's results. What follows is an arms race of continuous updates by the algorithm designers and continuous interrogation and manipulation of the proxies by the optimized entities. We examine the types of distortion that can be introduced, focusing on two web-relevant examples of optimization by proxy, and discuss potential alternatives.

Keywords
Algoritms, Optimization, Distributed Systems, Search

1. INTRODUCTION
The first thing a newly-hatched herring gull does is to peck on its mother's beak, which causes her to give it its first feeding. Puzzled by this apparent automatic recognition of its mother, Dutch ethologist and ornithologist Nikolaas Tinbergen conducted a sequence of experiments designed to determine what precisely it was that the newborn herring gull was attracted to [1]. After experimenting with facsimiles of adult female herring gulls, he realized that the beak alone, without the bird, would elicit the response. Through multiple further iterations he found that the characteristics that the newborns were attracted to were thinness, elongation, redness and an area with high contrast. Thus, the birds would react much more intensely to a long red stick-like beak with painted stripes on the end than they would to a real female herring gull. Objects of this class, able to dominate the attention of an animal away from the original target were later called 'supernormal stimuli' and have been commonly observed in nature ever since.

This is interesting from a web science perspective as a similar pattern appears when algorithms intended to make optimized selections over vast sets of candidates are applied on social systems. The algorithms' designers substitute the problem of measuring a characteristic hard to quantify such as relevance or quality, with a proxy which is computationally efficient to measure. As long as the fundamental assumption that the proxy indicates the presence of the desired property holds, the algorithm performs as intended, yielding results that to the untrained eye seem 'magical'. Google's PageRank, in its original incarnation [2], trying to optimize for page quality, optimizes for it indirectly, by analyzing the link structure of the web. As the web has grown, such algorithms, and their scalability characteristics, have helped search engines dominate navigation on the web over previously dominant human-curated directories. Similar algorithms are used in social news sites, recommendation engines and fraud detection systems.

2. EFFECT OF OTHER ACTORS
When there is only a single party involved in the production, filtering, and consumption of results, or when the incentives of the relevant group of actors are alligned, such as in the herring gull case, the assumption of the algorithm remains stable and its results remain of high quality.

In distributed systems such as the Web however, when the proxy is in the control of intelligent actors that can manipulate the proxy attribute, and they stand to benefit from distorting the results of the algorithm, then the existence of the algorithm itself and the motive distortions it creates alter the results it produces. In the example of PageRank, its early results owe to the fact that the link structure they crawled was effectively a byproduct of the buildup of the web. By bringing it to the attention of website owners as a distinct concept however, they have incentivized them to manipulate it separately, through techniques such as link farming and keyword stuffing, effectively making the altered websites act as supernormal stimuli for the algorithm. In this sense, the act of observation and the publication of results alters that which is being observed. What followed was an arms race between the algorithm designers and the external agents, each trying to affect the algorithm's results in their own preferred direction, with the algorithm designers controlling the algorithm itself and the external agents controlling part of the data it is applied on.

Web Science Conf. 2010, April 26-27, 2010, Raleigh, NC, USA.

3. DISTORTIONS INTRODUCED

One approach by algorithm owners is to keep secret the operation of the algorithm, creating uncertainty over the effects of manipulation of the proxy. This is effectively security by obscurity and can be counteracted by dedicated interrogation of the algorithm's results. In the case of PageRank, a cottage industry has formed around Search Engine Optimization (SEO) and Search Engine Marketing (SEM), essentially aimed at improving a website's placing in search engine results, despite the secrecy of the algorithm's exact current operation. While a distinction can be made between black-hat and white-hat practitioners, the fact remains that the existence of these techniques is a direct result of the existence of an algorithm that optimizes by proxy. Another trend related to the mechanics of search engine algorithms is what is called Googlebombing, where typically a grass-roots movement decides to link to a certain page using a predetermined keyword in the link text. This results in searches for that keyword returning the targeted site. The more successful the campaign, the higher in the results the target website appears. Googlebombing is used as a form of Internet graffiti, creating an artificial association between a keyword and a website. As a response to the various distortions, algorithms are enriched with heuristics to identify them. This, as the arms race progresses, converges to the point where the proxy approaches the original target more and more, and hence the external actors are forced to simulate the algorithm's target quality to the point where "sufficiently advanced spam is indistinguishable from content".

This of course would hold only if processing power were not an issue. However, if performance was not an issue, far more laborious algorithms could be used to evaluate the target attribute directly. Optimization by proxy, being a computational shortcut, is only useful when processing power is limited. In the case of the Web search, there is a natural asymmetry, with the manipulators able to spend many more machine- and man-hours to optimization of the result than the algorithm can spend judging the quality of a single item. Thus, algorithm designers can only afford to tackle the most broadly-occurring and easily distinguishable forms of manipulation, while knowingly ignoring the more sophisticated or obscure ones appearing at the long tail.

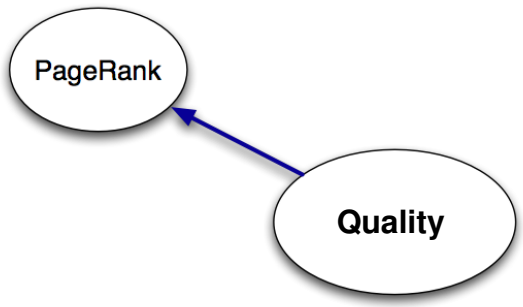

Figure 1. PageRank's role as a proxy makes it an indicator for the underlying cause.

4. FAKING IT – A BAYESIAN VIEW

We can obtain a little more insight by considering a simple Bayesian network representation of the situation. A key guide to algorithm design is the identification of some measure that intuitively will be highly correlated with quality. In terms of PageRank in its original incarnation, the reasoning is as follows. High quality/highly relevant web sites will attract attention from peers who are also contributing related content. This will "cause" them to link into the web site under consideration. Hence if we measure the number of highly ranked web sites that link into it, this will provide us with an *indication* of the quality of that site.

The key feature is that the causal relationship is *from* the underlying quality (relevance) *to* the indicator that is actually being measured (see figure 1).

This simple model raises a number of issues with the use of proxies. Firstly, one needs to be aware that it is not just a matter of designing a smart algorithm for quantifying the proxy. One also needs to quantify the strength of association between the proxy and the underlying concept.

Secondly, unless the association is an extremely strong one, this makes use of the proxy a relatively "lossy" test for the underlying concept. In addition, if one is going to use the proxy for decision-making, one needs some measure of confidence in the value assigned to the strength of the relationship – a second-order probability that reflects the level of experience and consistency of the evidence that has been used to determine the strength of the relationship.

Finally, one needs to be aware of the consequences of performing inference in the reverse causal direction. In modeling this as a Bayesian Network, we would use the conditional probability distribution p(PR | Q) as a measure of the "strength" of the relationship between cause and proxy (where "PR" is a random variable representing the value of PageRank, and "Q" is a random variable representing the value of the (hidden) cause, Quality). Given a particular observation of PR, what we need to determine is p(Q | PR) – the distribution over Quality given our observation on the proxy. This (in our simple model) can be determined through the application of Bayes' rule:

$$p(Q|PR) = \frac{p(PR|Q)\,p(Q)}{p(PR)}$$

What this is reminding us of us that the prior probability distribution on Quality is a major factor in determining its posterior following an observation on the proxy. Remember the textbook "clinician's fallacy" – a positive test for a rare disease actually leaves a low likelihood that the patient is suffering from the disease through the dominance of the prior (e.g, [6]).

Can this kind of model help us? Potentially it can if we are able to obtain some form of feedback or evaluation of the performance of a proxy. Well known learning algorithms can be used to update the probability distributions in such a network (see [7] for a simple introduction, or [8] for a much more technical review). In addition, one can place monitors on the rate of convergence of the probability distributions to see if there is a statistically significant disagreement between the predictive power of a proxy and the actual evaluated outcome in terms of quality (e.g. [9]). We need to perform a little more analysis but this ought to provide a relatively effective basis upon which to discount a proxy should some form of "stuffing" become prevalent for a particular proxy.

One other way forward is to explicitly model the use of multiple proxies using a Bayesian approach. To do this properly, one would need to include in the model a representation of the distributions (second order probabilities) over possible outcomes, in order to be sure that proxies that are being faked are suitably discounted. However, this does then turn the relative weakness of the proxies as predictors into a strength, as the impact of the fakes will be relatively rapidly discounted, whilst the remaining proxies can collaborate to provide much higher confidence in a prediction of Quality than any one on its own.

As should be clear from the age of many of the references in the above, none of this is new material. However, we do believe that it is important to bring this material into scope when discussing algorithms for optimizing selections.

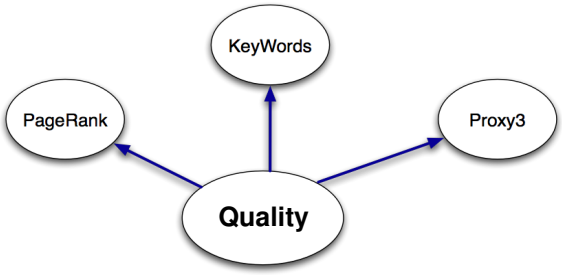

Figure 2. BN model using multiple proxies. Such a model will naturally adapt to use the most informative proxy if it can be combined with a feedback process. Note also that the indicators/proxies are marginally dependent, but conditionally independent given the true value of Quality. This again emphasizes their relative weakness as predictors if used in isolation.

5. PATHS TO A SOLUTION

Whereas algorithms have been introduced to the web as a way of taking the weight off humans, eventually we still have to supervise the algorithms, to continuously align the internal assumptions with the execution context, and interpret results appropriately. This, however, still means that many false positives and false negatives are required until a distortion is noticed by the supervisors, and there is a computational limit to the countermeasures that can be applied. Potential improvements include the consideration of costly signals[3], the use of communities with better aligned incentive cultures that encourage quality (such as Wikipedia), or the use of casual games as a way to provide vast training datasets for the algorithms [4]. However, as long as algorithms attempt to construct objective answers with a less than precise internal model of what constitutes a good answer, and as long as there is a strong enough incentive for adversaries to skew the results, the arms race will continue.

Looking at newer models that cover the above discussed are utilised to cover use cases of web search and news, often when traditional means fail, a radically different model seems to be emerging. To see it we need to look beyond the services that traditionally offer these services. Relative newcomers such as Q&A sites, social networks, and social search platforms provide alternative avenues for users to find news and answers.

Focusing on the distribution of news items in social networks such as Facebook and Twitter, users share the items that are important to them with others that follow them. Re-broadcasting features such as Twitter's 're-tweet', first appearing as a community norm and then formalized in the platform, allow diffusion to happen much more naturally. This is a radically different model than the top-down approach used by traditional media and even social news aggregators such as Digg and Slashdot. When results are determined for each user according to the sources of authority that the user trusts, and with an appropriate feedback loop for returned results, actors who contribute distorted information are inevitably sidelined, effectively rendering these distributed networks immune to optimization by proxy. This natural resistance to spam is in stark contrast to the huge amount of effort that goes into avoiding it in centralized aggregators, with varying results.

What this indicates in terms of optimization by proxy, is that a potential antidote is distributing the filtering logic. In the centralized single-algorithm paradigm, filtering is done through a single though elaborate model, uniformly applied to billions of pages. In its distributed counterpart, each user filters the news items they come in contact with, and shares them, with the results of the sharing being felt on the users' own reputation. Effectively there are millions of algorithms applied on the same data, without a large single target for manipulators to aim at. What's more, the judges, instead of computationally constrained algorithms, are actually well-motivated humans. In search, new entrants such as Aardvark, recently acquired by Google, have taken such a route.

These efforts are relatively new, and there is no clear evidence that any of them has been built with avoiding the effects optimization by proxy in mind, or even to serve the use cases we examined, as their stated purpose is different. To that extent, centralized and decentralized features often coexist, with the services being vulnerable to optimization by proxy to the extent that they introduce vulnerable features. One such example is Twitter's Trending Topics feature which attempts to algorithmically determine which topics are currently talked about throughout the network. As soon as a topic is uniformly broadcasted as 'trending', users are drawn to viewing the ongoing discussion, and spammers are rewarded for adding to the discussion keyword- and hashtag-laden advertising messages. This continues until the advertising stifles actual discussion, real users abandon the topic, and it stops trending. The cycle then restarts with a new set of trending topics. Q & A sites such as stackoverflow.com present another interesting hybrid, combining social dynamics and voting with the goal of creating answers of objectively high quality.

In all these examples, to the extent that distribution of filtering is applied, the trade-off for this new-found effectiveness is the increased burden to initially bootstrapping the network, as well as giving up the common point of reference that common result sets provide. However, with multiple people or algorithms to compete with, malicious content creators have less of a single target to manipulate, and therefore are forced to chose between getting ever closer to quality content in order to drive traffic, or giving up, both positive outcomes for the users.

6. CLASSIFYING THE APPROACHES

At this point, it may be helpful to pull together and classify the examples discussed so far, as to make the distinctions clear (see Table 1). The manual approach is the oldest one, with a group of human editors curating results for a much larger audience. The algorithmic approach is more modern, and in certain areas, such as search, has largely displaced the manual approach. It enjoys the benefits of greater reach and greater responsiveness, side-effects of its inherent scalability. However, we argue that it is vulnerable to a systemic fault in optimization by proxy. The effects of this weakness are not immediately obvious but tend to degrade results over time. A potential solution, by way of drastic paradigm shift, can be seen in new distributed alternatives. Here, distribution does not apply to the actual infrastructure, but rather to the filtering logic, which is handled directly by the end users. Each users' decisions have only a local effect, depriving attackers of a large and inflexible target to aim at.

Table 1. Classifying the various approaches to filtering

	Search	**News**
Manual	Human-curated Directories (Yahoo Directory, Alta Vista)	Traditional News Media (newspapers, TV)
Algorithmic	Algorithmic Search (Google PageRank)	Social News Aggregators (Slashdot, Digg, Hacker News)
(Hybrid) Distributed	Social Search, Q&A (Aardvark, Yahoo! Answers, Stack Overflow)	Social Networks (Twitter, Facebook)

7. CONCLUSION

While most teams that work in large-scale algorithmic systems are intimately aware of the characteristics of this pattern, we have not seen it being explicitly identified and analyzed before at this level of abstraction. Our aim in this paper has been to surface the concept as a stand-alone term, offering a common point of reference for its many instances, and hopefully sparking further discussion around its characteristics, proposed countermeasures, and fields of applicability. We have focused on the web-relevant examples of search and news, but as illustrated by the herring gull example in the introduction, it can be seen almost everywhere. Employers judging potential employees by the name of the university they attended, companies rewarding staff, especially in sales, with a productivity bonus, even academic funding bodies allocating funds according to bibliometrics, are only a few examples, and are all vulnerable to optimization by proxy. While highly experienced implementors in all fields get to know the pattern through experience, newer entrants have to rediscover its ins and outs through new real-world failures. By exposing this common pitfall, we hope to pave the way for future work in managing the problem for those who have no other option but to face it. One potential avenue may be a toolkit that packages the most successful strategies for applying optimization by proxy, monitoring its effectiveness, and potentially recognizing when it is no longer an effective strategy altogether.

8. ACKNOWLEDGMENTS

This work has been supported by the European Commission through IST Project OPAALS: Open Philosophies for Associative Autopoietic Digital Ecosystems (No. IST-2005-034824).

REFERENCES

[1] Tinbergen, N., Perdeck, A.C., 1950, “On the Stimulus Situations Releasing The Begging Response In The Newly Hatched Herring Gull Chick”, Behaviour, 3, 1-39.

[2] Page, L., Brin, S., Motwani, R., Winograd, T., 1998, “The pagerank citation ranking: Bringing order to the web”, Technical Report. Stanford InfoLab.

[3] Zahavi, A., Zahavi, A., 1997, “The Handicap Principle: A Missing Piece of Darwin’s Puzzle”, New York: Oxford University Press.

[4] von Ahn, L., “Games With A Purpose”. IEEE Computer Magazine, June 2006. pp 96-98.

[5] Jøsang, A., Hayward, R., Pope. S., “Trust Network Analysis with Subjective Logic”, Proceedings of the Australasian Computer Science Conference (ACSC'06), Hobart, January 2006.

[6] Elstein, A.S. and Schwartz, A, “Clinical problem solving and diagnostic decision making: selective review of the cognitive literature”, British Medical Journal, 2002;324:729-732.

[7] Krause, P.J., “Learning Probabilistic Networks”, Knowledge Engineering Review, 13: 321-351, 1999.

[8] Heckerman D., Geiger D. & Chickering D. “Learning Bayesian Networks: The Combination of Knowledge and Statistical Data”. Machine Learning, 20, 197-243, 1995.

[9] Spiegelhalter D.J. & Lauritzen S.L., “Sequential Updating of Conditional Probabilities on Directed Graphical Structures”. Networks, 20, 579-605, 1990.