How to keep bad papers out of conferences (with minimum reviewer effort)

Jonathan Anderson, Frank Stajano and Robert Watson
{firstname.lastname}@cl.cam.ac.uk

March 11, 2011

Abstract

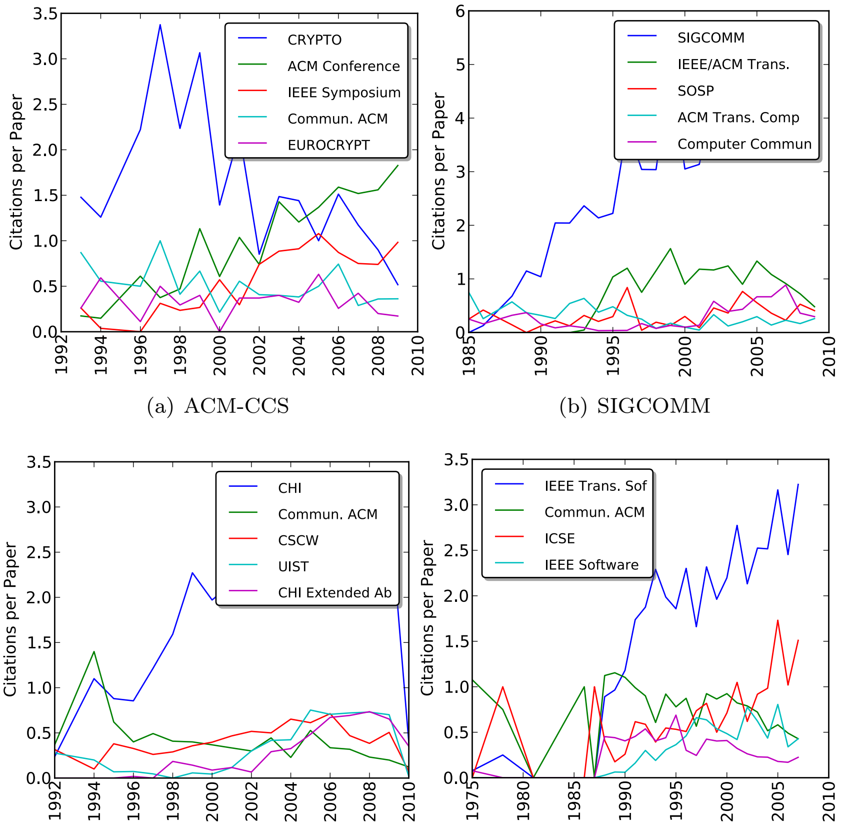
Reviewing conference submissions is both labour-intensive and diffuse. A lack of focus leads to reviewers spending much of their scarce time on papers which will not be accepted, which can prevent them from identifying several classes of problems with papers that will be. We identify opportunities for automation in the review process and propose protocols which allow human reviewers to better focus their limited time and attention, making it easier to select only the best "genetic" material to incorporate into their conference's "DNA." Some of the protocols that we propose are difficult to "game" without uneconomic investment on the part of the attacker, and successfully attacking others requires attackers to provide a positive social benefit to the wider research community.

1 Motivation

Figure 1 shows the trend of some research communities built around publication venues—both security-centric and not—to cite their own work more and more over time, to the exclusion of "outside" research[1]. We hypothesise that, as communities become increasingly introspective, they might delve deeper into the depths of the problems they care about and lose the continual exposure to new "genetic material" (problems and ideas) that is essential for the health of any community. It is important for research communities to optimise existing solutions to existing problems, but if authors only submit "the kind of paper that always gets in," if new ideas and new problems are never explored, the overall well-being of the community may suffer—their work may cease to be cited by other communities (their "genetic material" may not be selected for by discerning partners).

Similarly, we believe that authors may attempt to "mate" with a conference without proving their "genetic fitness" by abusing citations or coasting on reputation. Citations are meant to be a signal that an author has read

[1] All graphs in this paper have been generated from data supplied by arnetminer.org [10]. The data set consists of papers known to DBLP, linked together by a citation graph built by the ArnetMiner team, and "cleaned up" by the author. Despite this cleanup process, there are holes in the data, and erroneous data remains (e.g. citations of a technical report from the previous year turn up as citations of the journal article version, from two years in the future).



(a) ACM-CCS

(b) SIGCOMM

(c) CHI

(d) IEEE Trans. on Soft. Eng.

Figure 1: Inter- and intra-conference citations.

and understood the work of others. Unscrupulous authors, however, can tailor them to the program committee at little cost—citing what they've written [2, 1, 3, 4, 5, 6, 7, 8, 9, 11], citing what they like, citing what will make them think the authors are very well versed in the literature, all without actually reading and understanding the work they reference. Established authors, who have previously published at a conference, can also send the kind of signals that they know will be well-received, whereas proving one's fitness to a new research community may require truly superior research in order to overcome that community's institutional resistance to change.

We believe that we can improve on all of these problems via the judicious application of automation. In the context of a proposed threat model (Section 2), we propose that reviewers apply statistical techniques to measurable things in order to focus reviewer energy on unmeasurable things. We can remove some labourious work from reviewers' shoulders (Section 3.1): we can force attackers to step up their game, ultimately writing better papers or, if attempting to game the system, provide useful benefits to the wider research community (Section 3.2). We can look for authors attempting to socially engineering themselves into acceptance (Section 3.3), and we can promote diversity of ideas within conferences, reducing monocultures and increasing "genetic" health (Section 3.4).

2 Threat Model

We consider the defence of sacrosanct publication venues—conferences, journals, workshops—by dedicated but very busy Program Committees. These defenders have limited time with which to repel the invading hordes of PhD students[2], all trying to get their work published with minimal effort and literacy in the ancient lore of the discipline.

We consider that attackers can submit "the kind of paper that always gets in to this conference", attempting to start a vicious cycle of monoculture begetting monoculture, leading to the ultimate stagnation of the conference.

We believe that attackers can cite work which the Program Committe wrote or otherwise likes, without paying heed to—or even reading—the work, because it is very expensive for the reviewer to verify a citation, and she has to do it many, many times.

We recognise that attackers, in current systems, can distract reviewers with "busy work," keeping them from shepherding good papers and detecting subtle but serious flaws (such as evil citations) in bad ones.

3 Mechanical Assistance

Reviewers' time is a scarce resource, which a conference review process should spend carefully. Assuming that semantically-meaningful content of submissions can be reliably extracted (e.g. submission requires latex sources as well as PDF files, or PDF-to-text is reliable), thre are several ways in which mechanical assistance can be provided to program committees. Such assistance will allow reviewers to focus their scarce attention on the aspects of the review process which cannot be automated—evaluating the quality of ideas and research.

3.1 Clustering Submissions

Today, some conferences ask authors to provide keywords that describe their work. Where employed, this scheme makes it easier for review tasks to be distributed among reviewers. Nonetheless, significant human effort is involved in sifting through e.g. all of the abstracts tagged "committment schemes." We propose that mechanical analysis of submissions' reference lists could augment this process.

We presume that submissions can be usefully classified by their citations: papers that cite similar work are likely to be about similar topics. Rather than asking reviewers to review a set of abstracts, then, reviewers might be asked to rank their familiarity with existing classic papers, likely a less onerous task.

Furthermore, the results of the classification can be compared with author-supplied keywords. Submissions whose keywords do not match the citation-based classification can then be flagged as "interesting" for one of two reasons:

[2] Note that two of the three authors are PhD students; determining the position of tongue relative to cheek is left as an exercise for the reader.

either the authors are using keywords in a clueless manner, or they have something interesting to say which defies the existing literature. Either way, flagging up a small portion of the total submissions for "interestingness" checks could be a much better use of reviewer time than trawling through vast oceans of submitted abstracts.

3.2 Signalling that Authors Possess What They Cite

A paper's citations are like a bird's plumage, enhancing the chances of the subject to be selected by a discriminating audience with a large field of suitors. Given the stakes, there are obvious motivations to exaggerate one's citations: it makes the authors appear literate, lending credibility to the submission; it may flatter members of the Program Committee [2, 1, 3, 4, 5, 6, 7, 8, 9, 11]; it may be seen as a prerequisite to working in the field ("you can't publish here unless you cite Smith's seminal work on Public Key Widgets"). Such behaviour dilutes the quality of the conference in the long run, however: it fills reference lists with meaningless data, reducing the amount of information per page of proceedings.

In order to discourage such behaviour, we propose an information signalling protocol. This protocol requires authors to signal that, at minimum, they have gone through the trouble of *finding* and *looking at* (though not necessarily reading thoroughly) everything that they cite. The protocol requires little reviewer effort and communications overhead, and can even be verified after publication by any third party who can read the bibliography. The protocol is in-band, relying on the annotation of citations, but authors cannot simply replay annotated citations from other papers.

Such a signalling protocol cannot *guarantee* that an author has, in fact, read and understood everything they cite. If we assume, however, that the "energy gap" between opening a PDF and actually skimming its conclusion is not large enough to overcome PhD students' genuine interest in learning, this should be a very useful signal.

In the reference list at the end of a submission, we require authors to annotate each citation with a single word within square brackets. This word is taken from the referenced document, a response to a challenge which is public but unique. This challenge, c_i (the challenge associated with reference i), is given by:

$$c_i = \left(n_i \bmod P, n_i \bmod N\right) | n_i = h\left(h_m\left(s\right) | h_m\left(i\right) | n_c\right) \tag{1}$$

where n_i is a nonce specific to a particular reference in a particular submission, P is the number of pages in the referenced document and N is a number smally enough to be easily counted by humans (e.g. less than 50). $h_m\left(x\right)$, the "metadata hash" of paper x, is defined as:

$$h_m(p) = h\left(n_c | \text{authors}\left(p\right) | \text{conference}\left(p\right) | \text{year}\left(p\right)\right) \tag{2}$$

where n_c is a nonce generated by the conference (perhaps the filename of a supplied LaTeXclass), $\text{authors}\left(p\right)$ is a comma-separated list of last names of the paper's authors (in the same order as on the paper itself), etc. Including $h_m\left(s\right)$,. the metadata hash of the submission, ensures that authors cannot collaborate in generating hashes[3]. Furthermore, if the authors have a partial list of words in

[3] That is, unless they collaborate to disseminate the relevant literature to researchers who haven't read it, which is surely a positive outcome!

the referenced document, the only way to change the challenge to a "favourable" value is by changing the author list—a high price to pay.

Verifying the responses associated with references should be a relatively low-effort task: most citations can be automatically examined, assuming the PC software has access to a large corpus of literature. Any "cache misses" (whether due to not having access to literature or poor PDF-to-text conversion) can be probabalistically flagged for human review, and the outcome will be a more complete corpus with better textual equivalents.

Finally, attackers could collude to build a large corpus of relevant literature with high-quality PDF-to-text conversion, but if they did, would it be such a bad thing? Surely such a corpus would be of benefit to the research community, although copyright holders may not be pleased[4].

3.3 Checking that Authors Have Read What They Cite

A more difficult, and therefore more interesting, problem is checking that authors not only possess what they cite, but have read and understood it, critiquing it or allowing it to influence their own research. This is clearly the province of the experienced human reviewer, but mechanical assistance could be provided to reviewers to help them focus their energy where it would be most productive.

Identifying Usage To fully combat the problem of "token citations", reviewers could look at every reference in every submission and ask themselves, "can I see the influence of the referenced work on the submission?" In the real world, such analysis is impossible due to time constraints. Software tools could help reviewers, however, by displaying every place a reference is cited, including a few additional lines of context. Such a tool, especially if employed primarily for "interesting" references (see below) could help reviewers make cursory inspections and quickly judge whether or not the cited work has an influence on the submission.

Identifying Outliers One class of "interesting" references which are trivial to identify is that of statistical outliers. Figure 2 shows the increasing age of references, most of which are outliers, at the ACM Conference on Computer and Communications Security. These outliers date back to some of the seminal papers in computer security, and may be important references whose lessons have greatly influenced authors' thinking. Unfortunately, they may also be token citations which merely provide an air of historical literacy. Sorting the wheat from the chaff clearly must be done by a human; identifying which references particularly need to be sorted can be done more effectively by computer.

Identifying PC Citations Another class of "interesting" references are those which have been written by Program Committee members. Clearly, PC members are more likely to be cited than the average author; if their work were not valuable to the field, they would not be on the committee! Nonetheless, authors may be tempted to pad long strings of citations [2, 1, 3, 4, 5, 6, 7, 8, 9, 11] with PC members in an effort to flatter them or "pay their dues." Such citations are

[4] Of course, an wise publisher would use such a corpus to improve their own PDFs and citation graphs, again, a useful service to the research community.

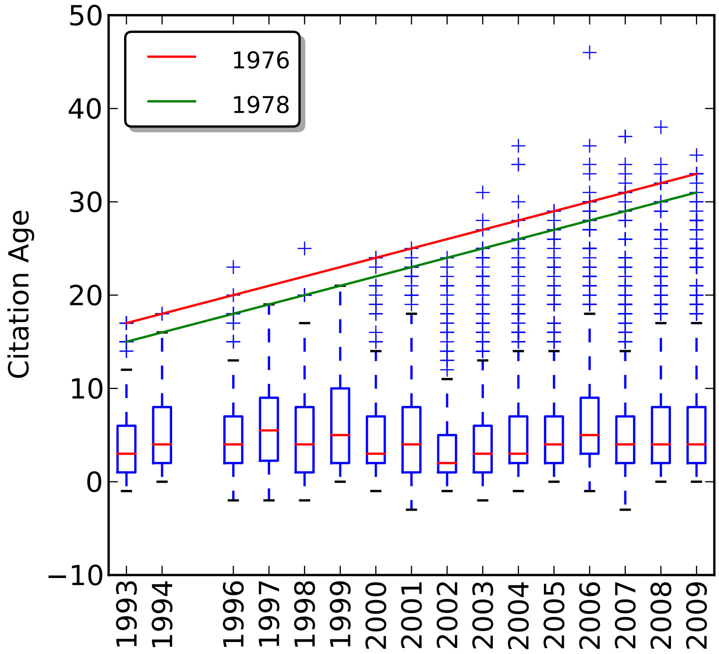


Figure 2: Citation Age at ACM-CCS.

therefore more interesting than many others, and thus can be probabalistically flagged for human review.

3.4 Encouraging Diversity

Human reviewers can often sort submissions very quickly into three bins: "definitely accept", "definitely reject" and "maybe." Much effort then goes into deciding which of the "maybe" papers deserve to be accepted, even though the number of papers to be accepted may be a small fraction of the "maybe" category. In such cases, automation can help provide two properties that we consider useful: focussing the most reviewer effort on a small number of to-be-accepted submissions and encouraging diversity within the conference.

From biospheres to computer security, monoculture is often recognised as a systemic weakness, but as stated above, we may be able to observe a worrisome trend towards monoculture in Figure 1. In order to encourage diversity, then, PC software could treat submissions preferentially that cite substantially different work from the papers that will definitely be accepted—if the reviewers have already accepted six papers on the finer points of zero-knowledge proof, perhaps one paper about a new real-world problem would be a breath of fresh air, an injection of new "genetic material," even if the sixteen zero-knowledge proof papers in the "maybe" bin are slightly better written.

One interesting property of this scheme is that, even though it is a statistical-classification–driven approach, it is very difficult to tactially adapt to: since its inputs are not "what got accepted last year", or even "what got submitted this year", but "what has been accepted this year", the kind of paper which will be most advantaged this year cannot be known until all of the first-pass reviews are in. The fact that social networks get an automated leg-up one year in no way implies that they will again next year: in fact, once they become a bandwagon, there will be pressure to get off the bandwagon and restore an interesting balance of work.

4 Future Work

We would like to conduct an experiment with a real program committee in which we ask each reviewer how much time they spent reviewing each submission, broken down into per-submission activities such as "reading up on things the submission cites", "convincing myself the idea works", "explaining why the idea doesn't work", "correcting grammar", etc.[5]

With more data, we would also like to explore the relationship between regular conference attendees and their publication records. Do authors who attend a conference every year tend to be more "introspective" than those who do not? Does attending a conference encourage others to cite your work, even if that work was not published at the conference in question? Only data can tell.

5 Conclusion

Through the judicious application of mechanical assistance, we believe that the conference submission review process can be made more efficient, focusing the limited time and energy resources of reviewers on those problems which can only be solved by humans. Furthermore, mechanical assistance could encourage good "genetic hygene" in conferences, leading to overall better health in the future.

References

[1] Christianson, B., and Harbison, W. S. Why Isn't Trust Transitive? *Lecture Notes in Computer Science* (1997), 171–176.

[2] Christianson, B., and Malcom, J. A. Binding Bit Patterns to Real World Entities. In *Proceedings* (1997), pp. 105–113.

[3] Crispo, B., and Christianson, B. A Note About the Semantics of Delegation. In *Proceedings* (1999).

[4] Lomas, M., and Christianson, B. Remote Booting in a Hostile World: To Whom Am I Speaking? *IEEE Computer* (1995), 50–54.

[5] Low, M. R., and Christianson, B. Technique for authentication, access control and resource management in open distributed systems. *Electronics Letters 30*, 2 (1994), 124–125.

[6] Stajano, F., and Anderson, R. The cocaine auction protocol: On the power of anonymous broadcast. *Information Hiding* (1999), 434–447.

[5] Obviously, such data could not be shared with the chair, otherwise reviewers might experience some trepidation about honestly expressing how much or how little time they spend on each paper.

[7] Stajano, F., and Anderson, R. The resurrecting duckling: Security issues for ad-hoc wireless networks. *Lecture Notes in Computer Science 1796* (2000), 172–182.

[8] Stajano, F., Wong, F., and Christianson, B. Multichannel protocols to prevent relay attacks.

[9] Švenda, P., Sekanina, L., and Matyáš, V. *Evolutionary design of secrecy amplification protocols for wireless sensor networks.* ACM, mar 2009.

[10] Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., and Su, Z. *ArnetMiner: extraction and mining of academic social networks.* ACM, aug 2008.

[11] Wong, F.-L., Stajano, F., and Clulow, J. Repairing the Bluetooth Pairing Protocol. *the Thirteenth International Workshop on Security Protocols (SPW) LNCS 4631* (2005).