Studying the source code of scientific research

Graham Cormode
graham@research.att.com

S. Muthukrishnan
muthu@cs.rutgers.edu

Jinyun Yan
jinyuny@cs.rutgers.edu

ABSTRACT

Just as inspecting the source code of programs tells us a lot about the process of programming, inspecting the "source code" of scientific papers informs on the process of scientific writing. We report on our study of the source of tens of thousands of papers from Computer Science and Mathematics.

1. INTRODUCTION

In understanding software artifacts, the source code is the primary subject of study. The open source movement emphasizes the availability of source code as a mechanism to allow extensibility and maintenance of software projects. Even within 'closed source' development, source code is typically accessible and readable to most users within version control systems. The study of source code helps to understand design decisions, and explains the detailed logic behind the final software object.

Meanwhile, in the world of research, the unit of production—the research paper—is typically presented in "compiled" form, say as a PDF file. However, this is also created from some underlying source code: a document that is processed to produce the PDF output. In contrast to the software setting, the source code here is typically guarded closely, and not even shared with close colleagues outside of the core authorship team. This practice is unfortunate, since the study of the source code of research papers has the potential to provide great insight into the process of communicating research work, and the differing norms and traits across fields.

The source code of a research paper can include many features that are either not present or hard to extract from the final object. For example, this can include multiple earlier versions of the text, internal comments and notes, and sections of text which are absent from the final version. It can also include descriptions of how to create complex formatting (e.g. tables) in a way that is easier to parse; internal labels for sections and references that are replaced by numeric references or hyperlinks in the final version, and so on. Such features can cast more light on the internal thought process that goes into producing the final paper, and allow automatic extraction of trends and patterns. In this article, we describe some of the observations we have been able to make in this direction, thanks to access to a large trove of research source code.

Digging in the arXiv. Our analysis is possible due to some properties of the arXiv technical report service. The arXiv is an open-access web-based e-print repository that covers many scientific fields, including physics, mathematics, nonlinear sciences, computer science, quantitative biology, quantitative finance and statistics[1]. Across all areas, over 700,000 documents have been made available via the service. The service began in 1991, and is primarily maintained and operated by the Cornell University Library. After registration, users may upload new documents, or revisions of their existing documents. A key feature is that arXiv **strongly** encourages users to provide source files for a paper, rather than the "compiled" version. If PDF generated from TEX/LATEX is detected, it is rejected, and the user is requested to provide source files instead.

Several upload formats are allowed, including TEX/LATEX, HTML, PDF, Postscript and (MS) Word. Our study focuses on Computer Science and Mathematics where TEX/LATEX predominates, and so forms the bulk of our discussion[2].

We used the API provided by arXiv to collect a large sample of papers in April 2011. We collected all 26,057 papers with the area of Computer Science, and 39,178 from the area of Mathematics (the set of subcategories ordered by their two character names in the range `math.AC` to `math.MG`).

2. FINDINGS

Based on our study of this dataset, we made a number of observations about the style and content of scientific writing in Mathematics and Computer Science, which would be either difficult or impossible to draw from the corresponding PDF files. Further details of our data collection and analysis are given in [3].

LATEX rules in the mathematical sciences. While a skilled reader can determine which tool was used to create a paper based on typographic peculiarities, it is challenging to automatically analyze a PDF to make this determination. Looking directly at the source files, we determined that over 87% of submissions to the arXiv in our dataset came from LATEX. This was even more stark when broken down by subject: 98% of papers under Mathematics are from LATEX source. Of the non-LATEX papers, the bulk are presented in PDF form. Examining the metadata of these, we found 70% contained the terms "Microsoft" or "Word", indicating that

[1]`http://arxiv.org`
[2]In what follows we refer to LATEX, with the understanding that this incorporates the TEX format.

the second choice tool in these areas is the Microsoft Word word-processor.

A secondary feature of how LaTeX is used is whether the input is structured into multiple files. Strikingly, 66% of Mathematics papers have a single .tex source file, indicating that the paper is fully self-contained: it does not call on any external figures or bibliographic files. In contrast, 82% of Computer Science submissions are formed of multiple source files.

Vocabulary choice. Given the source data, it is straightforward to extract all the words used in the corpus of scientific papers. We can then compare the relative frequency of different terms between such corpora. Comparing Computer Science and Mathematics, we can measure the occurrence frequency of words in each, and identify those that have the sharpest difference in usage. Perhaps most telling is the term that is used most differently in each: in Computer Science, the word that is used most in comparison to Mathematics is "algorithm", while in the other direction it is "equation". Arguably, this cuts to the heart of the difference in focus between the two fields.

The top-ten words that are used most often in Computer Science compared to Mathematics are

algorithm, time, figure, data, number,
state, model, information, probability, problem

while in the reverse direction, we obtain

equation, let, alpha, lambda, infty,
omega, frac, gamma, mathbb, map.

While these terms should be intelligible to researchers in either field, it is clear that notions such as "data" and "information", techniques such as "probability" and "algorithm" and concerns such as "time" are central to Computer Science. Meanwhile, the words that distinguish mathematical writing are primarily for common symbols: "alpha", "lambda", "gamma", "omega", "infty"; or for formatting in LaTeX, like "frac" and "mathbb".

Signs and symbols. LaTeX makes it easy to express mathematical symbols. It is instructive to study their relative occurrence. For example, we noted that the symbol for "less than or equal to" ($\leq$) is dramatically more common than "greater than or equal to" ($\geq$): there are 85% more occurrences of $\leq$. We suggest that this is indicative of a common expressive trope to bound a quantity of interest, and then provide subsequent upper bounds on this quantity that are progressively simpler to state. Comparing the number of open parentheses to the number of close parentheses, we observed that 0.7% of open parentheses are not closed. If anything, this is lower than we might expect.

Summertime, and the living is easy? The arXiv also provides metadata, such as the date of upload of a paper. The pattern of uploads is non-uniform throughout the year, with a pronounced drop during July and August, as illustrated in Figure 1). The number of papers uploaded in August is 16% below the average amount. These two months are when the majority of research universities in the northern hemisphere have their summer vacations. This seems at odds with the oft-repeated complaint of academics that they can't wait for the summer in order to get more research work done. However, we can reconcile these if we assume that while the research is performed at this time, it is not written up and circulated until later.

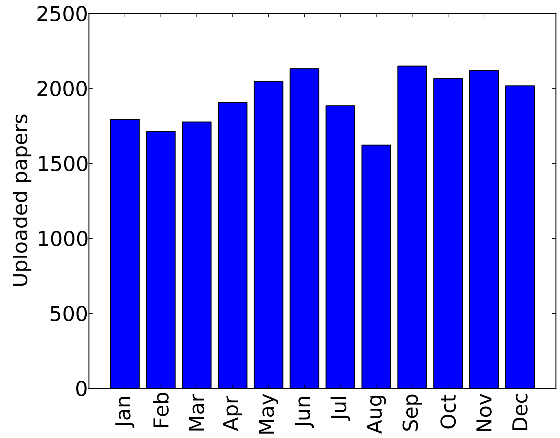
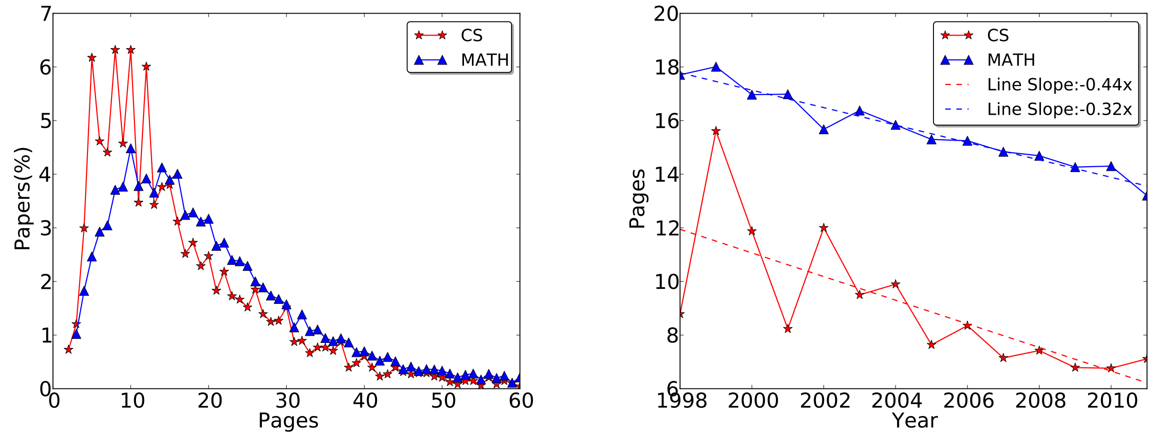


Figure 1: Number of CS papers uploaded to arXiv by month

You should probably take them out. Just as in program code, LaTeX code can include comments: unrestricted free text which does not alter the eventual output. This can be used for a variety of purposes, such as removing unwanted text, retaining a previous draft, communication between authors, commentary, notes and outlines. The arXiv FAQ includes the question "What if my TeX source has potentially embarrassing self-comments in it?" and provides the answer "Well... you should probably take them out. It is easy to strip these out in advance of submitting.". It also links to a script that will find and remove LaTeX comments created using the '%' symbol. This advice notwithstanding, we observed that 95% of Computer Science papers, and 90% of Mathematics papers included comments of some form. On average, CS papers had 772 words in comments, while in Math it was 395.

Based on a visual inspection, we determined that a majority of comments were essentially innocuous: containing redrafted text or LaTeX commands. However, there is also a non-trivial occurrence of more "sensitive" comments, in the form of discussion between authors discussing strategies for presenting results, expressing doubt as to the validity of proofs, and denigrating the work of others. The word usage within comments is somewhat different to that in the rest of the papers: words such as "latex, file, you, version" are the words with the biggest increase in usage. These are indicative of comments being used for communication and version control purposes.

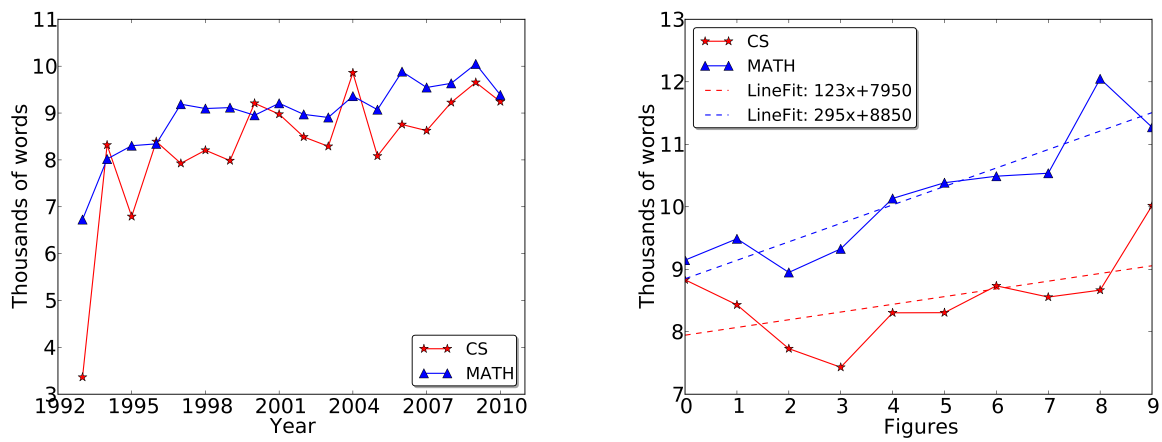
The sacred and the profane. Compared to everyday speech, the language of the research paper tends to have much fewer occurrences of oaths and curse-words. Nevertheless, they are not unknown. In some cases, apparently harsh language turns out to be just specialized terminology: the term "jerk" is used throughout the corpus primarily in the sense of a sudden change in acceleration, rather than in the derogatory sense. Likewise, "dumb" is used to describe simple-minded strategies, as a contrast to "smart" approaches.



(a) Distribution of paper lengths in pages

(b) Downward trend in pagelength over time

Figure 2: Page length distributions



(a) Upward trend in word length over time

(b) Paper length in words against number of figures

Figure 3: Trends with length in words

In other cases however, it seems that profanity is used internally to reflect the author's true feelings: there are examples where a particularly difficult example is given the (internal) label "bastard"; a macro for a complexity class is given the handle "crap"; and an initial theorem is labeled "bullshit" with an improvement provided. One particular notable example is of the occurrence of "bollocks" (a British-English idiom with broadly negative connotation) which occurs over fifty times within a single paper. Closer inspection reveals that this is because the central theorem in the paper is given the label "dogs-bollocks" and referred to extensively throughout; this phrase is a (coarse) British-English idiom with a strongly *positive* connotation. There are examples of profanity used in comments: the observation that "the \thanks layout looks crappy!"; the single word "bullshit" prefacing some technical text which has been commented out; and the comment "Who the fuck is —?" immediately after an acknowledgment to the named individual.

Size isn't everything. We also analyze the size of papers submitted to the arXiv, plotted in Figure 2(a). A first observation is that Mathematics is essentially unimodal in terms of pagelength, with a mode of 10 pages. Meanwhile, the distribution for Computer Science is more variable, with peaks at 5, 8, 10 and 12 pages. It is to be noted that many conferences Computer Science have page limits of this length, suggesting that many arXiv submissions correspond to conference submissions (uploaded in the conference format). When we study the trend of pagelength against time shown in Figure 2(b), there is a clear downward trend, which appears linear. Extrapolating this trend beyond the bounds of common sense, we obtain that the average Math paper will have no pages by 2052, while for Computer Science, this date will be 2026. However, when we study the behavior of length in terms of words, the trend is actually *increasing*: papers have more text over time according to Figure 3(a). Combining these two observations, the conclusion is that papers are being posted to arXiv in increasingly dense layouts.

Is a picture is worth 1000 words? The old adage, "A picture is worth a thousand words", suggests that adding illustrative figures should tend to reduce the length of a document. However, we observe the opposite trend from Figure 3(b): in both Math and CS, adding figures *increases* the length of a paper. In Math, the trend seems to be fairly consistent, and we have a new adage: "A pictures costs three hundred words". For CS, the trend is more variable, and weaker: the cost is an average of 120 words per figure. We might conjecture that in Math, figures are typically used to illustrate technical concepts which require some effort to describe, whereas in CS, many figures are data plots that need less text to interpret.

Performing the same calculation with the use of theorems to encapsulate central results, both CS and Math show a similar trend, which is very consistent: each theorem lengthens the paper by around 600 words. This makes sense: the statement, discussion and proof of a theorem should require some reasonable amount of additional text. Use of theorems is more characteristic of Math: at least 71% of Math papers contain a theorem, while only 48% of CS papers contain theorems. However, for papers with theorems, the distribution is not so different: CS papers have 4.85 theorems on average, while Math papers have 5.51.

3. CONCLUDING REMARKS

From these observations, we have demonstrated that there are many trends, patterns and behaviors in the writing of research papers that can be found by study of their "source code". In the description of our full study, we provide more examples, and motivate the development of the area of "scienceography", as the study of scientific writing [3]. This complements the much more substantial body of work that analyzes other facets of the scientific corpus, such as bibliometrics and scientometrics, which focus on citation patterns across papers [1; 2; 4]. While there has been detailed study of scientific writing in the past, such as considering the different forms of argument and persuasion used, these have tended to study on a single paper at a time [5; 6]. The hope is that the study of larger data sets can have an effect akin to the impact of online social network data on social network analysis.

Many future directions are possible. The natural direction is to expand the scope of this study, in terms both of the subject areas studied and types of data used in the analysis. In addition to LaTeX documents, there is relevant information with other formats of document (principally the Microsoft Word formats), and from other sources: say, multiple versions of a paper from a version control system; or in the (source code for) slides for a conference presentation or poster. Here, the first challenge is to identify suitable sources which can provide sufficient quantity of data to provide statistically meaningful insights.

Acknowledgments. This material is based upon work supported by the National Science Foundation under Grant No. 0916782.

4. REFERENCES

[1] Helen Barsky Atkins and Blaise Cronin. *The Web of Knowledge*. Information Today, 2000.

[2] Nicola De Bellis. *Bibliometrics and Citation Analysis: From the Science Citation Index to Cybermetrics*. Scarecrow Press, 2009.

[3] Graham Cormode, S. Muthukrishnan, and Jinyun Yan. Scienceography: the study of how science is written. In *Proceedings of FUN with Algorithms*, 2012.

[4] Henk F. Moed. *Citation Analysis in Research Evaluation*. Springer, 2011.

[5] Frederick Suppe. The Structure of a Scientific Paper. *Philosophy of Science*, 65(3):381–405, 1998.

[6] Steven Yearley. Textual persuasion: The role of social accounting in the construction of scientific arguments. *Philosophy of the social sciences*, 11:409–435, 1981.