Publishing Your Research

Daniel A. Jiménez
Texas A&M

Armando Solar-Lezama
MIT

CRA-WP
Computing Research Association
Widening Participation

Daniel A. Jiménez

- Education
  - BS/MS Computer Science, UT San Antonio 1992/1994
  - Ph.D. Computer Sciences, UT Austin 2002
- Jobs
  - Instructor/Research UT Health Science Center San Antonio
  - Assistant/Associate Professor, Rutgers
  - Associate/Full/Department Chair, UT San Antonio
  - Professor, Texas A&M University
  - 3 sabbatical leaves at research institutions in Spain (UPC & BSC)
  - Consult with industry
- Research
  - Computer architecture: front-end microarchitecture, cache management
  - Invented perceptron branch predictor currently in your PC or phone
  - IEEE Fellow, Bob Rau award for branch prediction
  - Very diverse group of Ph.D. graduates
- Personal
  - Dual citizen USA/México
  - Born and raised in Texas
  - Married with one daughter

CRA-WP
Computing Research Association
Widening Participation

Armando Solar-Lezama

- I was born in Mexico City
- My whole family moved to Texas when I was 15
- BS in Computer Science and Math: Texas A&M University 2003
- PhD in Computer Science: UC Berkeley 2008
- @MIT ever since where I lead the Computer Aided Programming Group and I am Associate director of CSAIL.

Publishing your research

- Step 1: Do some great research ✓
- Step 2: Write it up into a great paper
- Step 3: Get it published in a top venue

Writing a great paper

- A great paper needs to convey three things:
  - That you have accomplished something that had never been accomplished before.
  - That there is a new idea behind your accomplishment, that this wasn't just another turn of the crank.
  - How it connects to the broader literature.

Structure

- Introduction
- Overview
- Method
- Evaluation
- Related work
- Discussion/Conclusion

Introduction

- The three elements of a good paper need to be crystal clear in the introduction.
  - New accomplished, based on new idea, connected to the literature
- The introduction is a contract.
  - If the introduction says "my method is the fastest" then you better have a really solid performance evaluation.
  - If it says "my method improves the usability" then you better have a user study that actually evaluates usability.
  - If you say “My method can find bugs in real software" but you only tested it on synthetic bugs injected into small code snippets, then it's not going to fly.

CRA-WP
Computing Research Association
Widening Participation

Overview

- Sometimes part of other sections
- Build intuition
  - Use a running example
  - Favor intuition over precision
  - Examples:
    - What does your algorithm do on a concrete example?
    - What is it like to use your new interface?

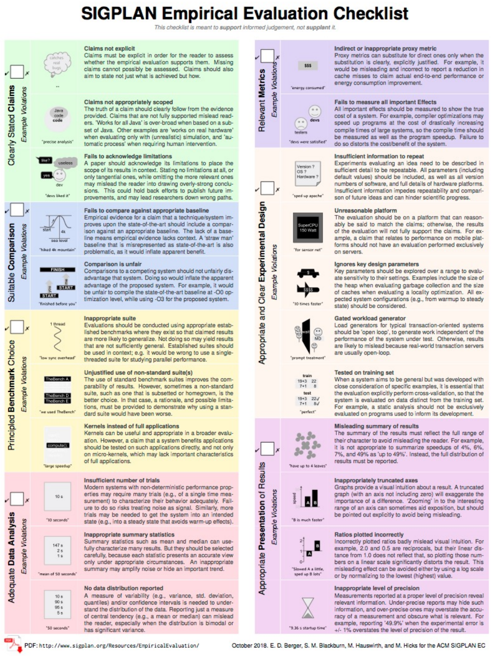
Method

- This is where you explain the details of what you did.
- Pitfalls:
  - This should not just be a code dump, or a text description of your algorithm. Break it into meaningful components, give them names.
  - Make sure you introduce every term before you use it.
  - Make sure the background is appropriate to the audience.
  - Make sure it's clear to the reader what's background and what you actually invented.

Evaluation

- Make it very clear what are the questions that the experiments are supposed to answer.
- It should be crystal clear that you went out of your way to try to disprove your hypothesis.

Evaluation



SIGPLAN Empirical Evaluation Checklist

*This checklist is meant to **support** informed judgement, not **supplant** it.*

Clearly Stated Claims
Example Violations

Claims not explicit
Claims must be explicit in order for the reader to assess whether the empirical evaluation supports them. Missing claims cannot possibly be assessed. Claims should also aim to state not just what is achieved but how.

Claims not appropriately scoped
The truth of a claim should clearly follow from the evidence provided. Claims that are not fully supported mislead readers. 'Works for all Java' is over-broad when based on a subset of Java. Other examples are 'works on real hardware' when evaluating only with (unrealistic) simulation, and 'automatic process' when requiring human intervention.

Fails to acknowledge limitations
A paper should acknowledge its limitations to place the scope of its results in context. Stating no limitations at all, or only tangential ones, while omitting the more relevant ones may mislead the reader into drawing overly-strong conclusions. This could hold back efforts to publish future improvements, and may lead researchers down wrong paths.

Suitable Comparison
Example Violations

Fails to compare against appropriate baseline
Empirical evidence for a claim that a technique/system improves upon the state-of-the-art should include a comparison against an appropriate baseline. The lack of a baseline means empirical evidence lacks context. A 'straw man' baseline that is misrepresented as state-of-the-art is also problematic, as it would inflate apparent benefit.

Comparison is unfair
Comparisons to a competing system should not unfairly disadvantage that system. Doing so would inflate the apparent advantage of the proposed system. For example, it would be unfair to compile the state-of-the-art baseline at -O0 optimization level, while using -O3 for the proposed system.

Principled Benchmark Choice
Example Violations

Inappropriate suite
Evaluations should be conducted using appropriate established benchmarks where they exist so that claimed results are more likely to generalize. Not doing so may yield results that are not sufficiently general. Established suites should be used in context; e.g. it would be wrong to use a single-threaded suite for studying parallel performance.

Unjustified use of non-standard suite(s)
The use of standard benchmark suites improves the comparability of results. However, sometimes a non-standard suite, such as one that is subsetted or homegrown, is the better choice. In that case, a rationale, and possible limitations, must be provided to demonstrate why using a standard suite would have been worse.

Kernels instead of full applications
Kernels can be useful and appropriate in a broader evaluation. However, a claim that a system benefits applications should be tested on such applications directly, and not only on micro-kernels, which may lack important characteristics of full applications.

Adequate Data Analysis
Example Violations

Insufficient number of trials
Modern systems with non-deterministic performance properties may require many trials (e.g., of a single time measurement) to characterize their behavior adequately. Failure to do so risks treating noise as signal. Similarly, more trials may be needed to get the system into an intended state (e.g., into a steady state that avoids warm-up effects).

Inappropriate summary statistics
Summary statistics such as mean and median can usefully characterize many results. But they should be selected carefully, because each statistic presents an accurate view only under appropriate circumstances. An inappropriate summary may amplify noise or hide an important trend.

No data distribution reported
A measure of variability (e.g., variance, std. deviation, quantiles) and/or confidence intervals is needed to understand the distribution of the data. Reporting just a measure of central tendency (e.g., a mean or median) can mislead the reader, especially when the distribution is bimodal or has significant variance.

Relevant Metrics
Example Violations

Indirect or inappropriate proxy metric
Proxy metrics can substitute for direct ones only when the substitution is clearly, explicitly justified. For example, it would be misleading and incorrect to report a reduction in cache misses to claim actual end-to-end performance or energy consumption improvement.

Fails to measure all important Effects
All important effects should be measured to show the true cost of a system. For example, compiler optimizations may speed up programs at the cost of drastically increasing compile times of large systems, so the compile time should be measured as well as the program speedup. Failure to do so distorts the cost/benefit of the system.

Appropriate and Clear Experimental Design
Example Violations

Insufficient information to repeat
Experiments evaluating an idea need to be described in sufficient detail to be repeatable. All parameters (including default values) should be included, as well as all version numbers of software, and full details of hardware platforms. Insufficient information impedes repeatability and comparison of future ideas and can hinder scientific progress.

Unreasonable platform
The evaluation should be on a platform that can reasonably be said to match the claims; otherwise, the results of the evaluation will not fully support the claims. For example, a claim that relates to performance on mobile platforms should not have an evaluation performed exclusively on servers.

Ignores key design parameters
Key parameters should be explored over a range to evaluate sensitivity to their settings. Examples include the size of the heap when evaluating garbage collection and the size of caches when evaluating a locality optimization. All expected system configurations (e.g., from warmup to steady state) should be considered.

Gated workload generator
Load generators for typical transaction-oriented systems should be 'open loop', to generate work independent of the performance of the system under test. Otherwise, results are likely to mislead because real-world transaction servers are usually open-loop.

Tested on training set
When a system aims to be general but was developed with close consideration of specific examples, it is essential that the evaluation explicitly perform cross-validation, so that the system is evaluated on data distinct from the training set. For example, a static analysis should not be exclusively evaluated on programs used to inform its development.

Appropriate Presentation of Results
Example Violations

Misleading summary of results
The summary of the results must reflect the full range of their character to avoid misleading the reader. For example, it is not appropriate to summarize speedups of 4%, 6%, 7%, and 49% as 'up to 49%'. Instead, the full distribution of results must be reported.

Inappropriately truncated axes
Graphs provide a visual intuition about a result. A truncated graph (with an axis not including zero) will exaggerate the importance of a difference. 'Zooming' in to the interesting range of an axis can sometimes aid exposition, but should be pointed out explicitly to avoid being misleading.

Ratios plotted incorrectly
Incorrectly plotted ratios badly mislead visual intuition. For example, 2.0 and 0.5 are reciprocals, but their linear distance from 1.0 does not reflect that, so plotting those numbers on a linear scale significantly distorts the result. This misleading effect can be avoided either by using a log scale or by normalizing to the lowest (highest) value.

Inappropriate level of precision
Measurements reported at a proper level of precision reveal relevant information. Under-precise reports may hide such information, and over-precise ones may overstate the accuracy of a measurement and obscure what is relevant. For example, reporting '49.9%' when the experimental error is +/- 1% overstates the level of precision of the result.

PDF: http://www.sigplan.org/Resources/EmpiricalEvaluation/

October 2018. E. D. Berger, S. M. Blackburn, M. Hauswirth, and M. Hicks for the ACM SIGPLAN EC

- SIGPLAN evaluation checklist

https://www.sigplan.org/Resources/EmpiricalEvaluation/

Related work

- Sometimes it goes at the end, sometimes it goes in the beginning.
- 3 categories of related work:
  - What you build upon,
  - what you compete with,
  - unrelated work

CRA-WP
Computing Research Association
Widening Participation

Discussion/Conclusion

- Your opportunity to discuss the implications of your work

General writing advice

Building Publishing Muscle

- Non-Archival Publications
  - Workshop papers
  - Poster Abstracts
  - Doctoral Symposia
- Archival Publications
  - Full-length Conference and Journal Papers

Know Your Audience

- Read lots of papers from the target venue
- Attend the venue (if a conference)
- Review for the venue if possible (ask your advisor to recommend you for this)
- Program Committee meetings
  - Senior students may get invited if their advisor pulls strings
  - You may be able to observe as a student volunteer

CRA-WP
Computing Research Association
Widening Participation

Make an Outline

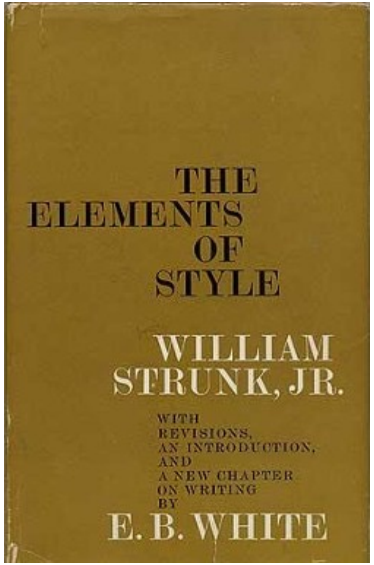
- Iterate and agree on the outline with your advisor before you start writing
- You don’t need to fill in the sections in order!
  - Sections I often find easier to write first: Related Work, Methods, Results
  - Sections I often save until later: Introduction, Discussion

Start Early

- The more iterations, the stronger the paper
- Set an internal deadline with your advisor
- When is a draft "advisor ready"? Perfection isn't expected
- Leave ample time for advisor and peer feedback, making submissions accessible, creation of video or other supplementary materials

CRA-WP
Computing Research Association
Widening Participation

Leverage Resources



- writing courses at your university
- reference books (Strunk & White)
- professional or pro bono proofreaders
  - Can you or your advisor apply for funding for this type of resource?
  - Free resources often include paper mentoring programs offered by conferences & professional societies

Getting it Published

Communicate with Co-Authors

- Agree on deadlines (for outline, drafts of sections, full draft, feedback, etc.)
- Agree on division of labor
- Be explicit about authorship (who & in what order)

Pick a venue

- Go for the best venue that works for your paper
  - but make sure it's a good match

About Deadlines

- What to do if the submission site crashes near the deadline…
- When is it OK to request an extension?

Metadata Matters

- Abstract Pre-Registration
- Keywords = Reviewer Matching
- What name should you publish under?

Rebuttals

- Sleep on it!
- What if your scores are very low or high?
- Prioritize reviewers’ comments & group by themes
- As with all writing, start early, get feedback, iterate
- More at aka.ms/rebuttals

CRA-WP
Computing Research Association
Widening Participation

Things to Avoid

- Plagiarism (including self-plagiarism)
- Dual submissions
- All-nighters (start early, iterate often!)
- Complaining about reviews on social media
- Submitting without knowledge of advisor/co-authors

A few parting tips & reminders

- Publications stay on your CV forever
  - Submit work you are proud of to venues you respect
- Be explicit and generous when determining authorship – and do it early on, it will only get more awkward with time
- Many things vary depending on area
  - authorship order (by contribution, convention, position)
  - #papers, conferences vs. journal, acceptance rates
- Reviews – learn from them and improve your work
  - When writing reviews yourself, imagine the authors reading them

CRA-WP
Computing Research Association
Widening Participation

Acknowledgments

- Thanks to Merrie Morris and Nancy Amato for their slides from last year

CRA-WP
Computing Research Association
Widening Participation

Questions?

CRA-WP
Computing Research Association
Widening Participation