

HOW TO READ A RESEARCH PAPER

Stacy
Patterson
10/12/2016

WHY READ A RESEARCH PAPER?

- To learn more about a topic
- To learn about work that is related to your research

- Because your advisor told you to
- Because you have to present the paper in a class
- Because you have to review the paper for a conference or journal
- Because a reviewer told you to cite the paper in your work

TYPES OF PAPERS (VENUE)

■ Conference Papers

- Peer-reviewed
 - Several Program Committee Members rate the paper
 - Highest rated papers are accepted
- Usually 3 to 6 months between submission and acceptance notification
- Paper is presented in a conference talk
- Published in conference proceedings

■ Journal Papers

- Usually longer than a conference paper
- May be based on a conference paper
- Peer reviewed: Several reviewers may shepherd the paper through multiple revisions
- 6 months to 2 years or more between submission and publication

TYPES OF PAPERS (VENUE – 2)

■ Workshop Papers

- Generally shorter than a conference paper
- May contain preliminary or partial results
- Faster publication time than conferences
- Peer-reviewed, similar to a conference paper

■ Technical Reports

- Published by the authors
- Can appear on school web site, arXiv.org, ResearchGate, etc.
- Not necessarily peer-reviewed
 - May be a pre-print of a conference or journal paper
 - May be a paper that was never submitted to conference or journal
 - May even be a paper that was rejected by a conference or journal

TYPES OF PAPERS (VENUE – 3)

■ Magazines

- Published more often than conference proceedings
- May be peer-reviewed
- Aimed at a more general audience
- Example: IEEE Computer Magazine

TYPES OF PAPERS (CONTENT)

- **Technical paper**
 - New ideas or results (can be analytical, experimental, real-world system)
- **Survey/Tutorial paper**
 - Review of existing work on a topic
 - Usually hundreds of references
 - Ideally organized and compared in a useful way
- **Vision/Opinion paper**
 - Proposes new problems and/or research directions
 - Advocates solutions for existing and/or new research problems
- This talk will focus on how to read technical papers

HOW TO FIND A PAPER

- **Web search: e.g., Google scholar**
 - Finding the right keywords is an art
 - Become a power searcher
- **References section of another paper**
- **A useful paper list**
 - Survey papers
 - A course syllabus
 - A blog or personal web page
 - arXiv email alert service
- **Personal recommendations from colleagues**

GETTING READY

- It may take several hours to read a single paper
 - Leave yourself time to take breaks if needed
 - Try reading with a friend – check in after each paragraph or subsection to make sure you both understand
- Read critically, but with an open mind
 - Don't automatically accept everything as true/correct/the best solution
 - But do look for the strong points of the paper
- As you read the paper:
 - Take notes of important or confusing points
 - Write down any questions you have
 - If you need to look something up, do it
 - Wikipedia is your friend

BEGIN AT THE VERY BEGINNING

- What is the title?
- Who are the authors?
 - Which are professors? Which are students?
- Where are the authors from?
- Where was the paper published?
- When was the paper published?

- Read the abstract to get a basic idea of what the paper is about

- You should learn who the leaders are in your field, what they are working on, and where they publish.

THE TWO PASS APPROACH

- **First, skim the paper**
 - Skip anything that takes significant mental effort
 - Just get a basic idea of what is in the paper
- **After skimming, you can decide if you want to read the paper**
 - Is the paper relevant to your research?
 - Do you have enough background to understand the paper?
 - If not, read some other papers first
 - Does the quality of the paper seem reasonable?
 - If not, check citation count in Google Scholar
 - Or ask your advisor/mentor about the quality of the venue
- **If the paper passes the first pass, then do a second pass where you read in detail.**

READING THE PAPER: THE INTRODUCTION

- **Goals of the introduction**
 - Give motivation for the research topic
 - Define the specific research problem in the context of a broad topic
 - Explain the contributions of the research paper and why they are important
- **As you read the rest of the paper, keep in mind what the authors promised in the introduction:**
 - Did the authors convince you the problem is important?
 - Does the solution make sense? Is it explained well?
 - Does the solution adequately address the problem?
 - How do the authors demonstrate this?

PARTS OF A PAPER: PROBLEM DESCRIPTION

- A formal detailed description of:
 - The system model, including assumptions
 - The problem(s) under considerations
 - Properties of the desired solution
- Questions to consider:
 - Does the formal problem description match the informal description in the introduction?
 - Are the model and assumptions realistic?

READING THE PAPER: SOLUTION

- Description of the solution(s) to the problem(s)
 - Algorithms
 - Software
 - Hardware
- Questions to consider:
 - Does the solution solve the problem?
 - Are there any potential weaknesses with solution?
 - Does it tolerate errors or component failures?
 - Is it prohibitively expensive (computationally or financially)?
 - Does the solution scale?

READING THE PAPER: ANALYSIS

- Theoretical results about the problem and/or solution
 - Proof of correctness
 - Asymptotic analysis (Big-O)
 - Error bounds
- If you want to really understand the theoretical results, try to reprove them.
- Questions to consider:
 - How well does the theory match the claims in the introduction?

READING THE PAPER: EXPERIMENTS

- Empirical evaluation of the proposed solutions
 - May be done on real system or in simulations
 - May use real-world or synthetic data sets
 - Usually includes figures – you should read them carefully
- Questions to consider:
 - Are the results generated using realistic scenarios (data and system)?
 - Do the authors compare their solutions to other solutions in a meaningful way?
 - Do the results match the promises made in the introduction?

READING THE PAPER: RELATED WORK

- Description of prior research related to the problem(s) and/or solution(s)
 - Should highlight differences

- This can be a good source for more papers to read.

READING THE PAPER: CONCLUSION

- Summarizes the paper contributions
- Sometimes gives ideas for future work
 - Potential research topics?

POST MORTEM

- When you are done with the paper:
 - Go back and review the notes and questions you wrote down as you read.
 - Do you still have questions? Make a note of these
- Keep a log (journal, blog, diary) of the papers you read
 - Write a short summary of the paper (2 to 3 sentences)
 - Also write down any questions or suggestions you have that relate to your research
- You will read a lot of papers – a log will help you keep track of them

OTHER THOUGHTS

- Authors are not perfect. Neither are most papers
- Papers may contain mistakes
 - If something looks incorrect, it may be
- Some papers may be hard to read
 - If you don't understand a section, it may not be your fault
 - You just have to give it your best shot

- "Never read the original paper on X first. Instead read several later papers on what they say about X, get an idea of X and then read the original paper. Somehow the research community is much better in explaining ideas clearly than the original authors themselves." *Delip Rao*

DO YOU WANT TO KNOW MORE?

- If you want to read more about the paper topic:
 - References cited in related work.
 - Forward references – papers that have cited the paper

The screenshot shows a Google Scholar search results page. The search query is "MapReduce: simplified data processing on large clusters". The results are filtered by "Articles". The top result is "MapReduce: simplified data processing on large clusters" by J. Dean and S. Ghemawat, published in Communications of the ACM in 2008. The abstract describes MapReduce as a programming model for processing and generating large datasets. The text "Cited by 18718" is circled in red. The second result is "Map-reduce-merge: simplified relational data processing on large clusters" by H. Yang, A. Dasdan, R. L. Hsiao, and D. S. Parker, published in Management of Data in 2007. The abstract describes it as a programming model for scalable parallel applications.

Web Images More...

Google

MapReduce: simplified data processing on large clusters

Scholar About 24,600 results (0.17 sec)

Articles **MapReduce: simplified data processing on large clusters** [HTML] [usenix.org](#)
J. Dean, S. Ghemawat - Communications of the ACM, 2008 - dl.acm.org

Case law Abstract MapReduce is a programming model and an associated implementation for processing and generating large datasets that is amenable to a broad variety of real-world tasks. Users specify the computation in terms of a map and a reduce function, and the ...

My library Cited by 18718 Related articles All 444 versions Cite Save

Any time **Map-reduce-merge: simplified relational data processing on large clusters** [PDF] [duke.edu](#)
H. Yang, A. Dasdan, R. L. Hsiao, D. S. Parker - ... on Management of data, 2007 - dl.acm.org

Since 2016 Abstract Map-Reduce is a programming model that enables easy development of scalable parallel applications to process a vast amount of data on large clusters of commodity machines. Through a simple interface with two functions, map and reduce, this model

Since 2015

Since 2012

RESOURCES AND REFERENCES

- <http://www.cs.columbia.edu/~hgs/netbib/efficientReading.pdf>
- <http://www.cs.jhu.edu/~jason/advice/how-to-read-a-paper.html>
- <http://www.sciencemag.org/careers/2016/03/how-seriously-read-scientific-paper>
- Mendeley Reference Manager: <https://www.mendeley.com/>

THANK YOU

- Any questions?