How to Write an Independent Work Paper

AARTI GUPTA

APRIL 6, 2021

Before we dig into details …

- When to start writing?
  - NOW!

- How to view the writing-research process?
  - Iterative
  - Describing intermediate results yields ideas about other experiments to run, and other data to collect
  - Advisor feedback can iteratively improve both the report and the research itself

How about LaTeX?

- Don't need to use it, but…

- Easiest way to include figures, equations, make citations, cross references …

- Written by programmers for technical writing

- Runs on your computer or web
  - Check out: Overleaf.com

- Try it, you'll like it!
  - Template on iw.cs.princeton.edu at
    - https://www.cs.princeton.edu/sites/default/files/uploads/template-20160413.pdf
    - Files to do this at https://www.cs.princeton.edu/courses/archive/www-coursefiles/iw/IWreport.zip
  - Example at https://www.overleaf.com/read/vwbpbhswvntr

Outline from Web Site

- Introduction: Motivation & Goal

- Problem Background and Related Work

- Approach

- Implementation

- Results

- Conclusions

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- Approach
- Implementation
- Evaluation
- Conclusions and Future Work
- Bibliography
- Appendices

Abstract

- Optional

- Good idea

- Succinct!
  - Problem
  - Results

- Not
  - Notation
  - Background

Abstract: Example

"This paper details the design, development, and evaluation of a Kinect-powered application to facilitate the instruction of ballroom dance. The application uses the Kinect camera's skeletal tracking capabilities to teach and evaluate users through various ballroom dance positions and concepts, taking the form of a number of training modules that end with a game-like assessment portion. This application aims to fill a hole in ballroom dance instruction, providing the ease of access of self-study materials alongside the quality of instruction of live coaching."

- "Using Kinect To Learn How To Ballroom Dance" Michael Li, Fall 2015

Enhanced Outline

- Abstract
- ➢ **Introduction**
- Problem Background and Related Work
- Approach
- Implementation
- Evaluation
- Conclusions and Future Work
- Bibliography
- Appendices

Introduction

- Motivate the work
  - Why do we care?
    - Give examples of real-world problems
      - Concrete, specific examples that have occurred in the real world are the strongest motivators
      - Statistics on the prevalence of the problem can also provide evidence

- Define the problem you solve clearly
  - Don't make readers guess

- Be explicit about your contribution!

- Get to the point

Enhanced Outline

- Abstract
- Introduction
- ➤ **Problem Background and Related Work**
- Approach
- Implementation
- Evaluation
- Conclusions and Future Work
- Bibliography
- Appendices

Problem Background and Related Work

- Context of your work
  - What is known?
  - What is similar?
  - What is different?

- May list several works at once
  - "Several others have proposed approximations [a, b, c]."

- Summarize closest, most important to your work

- **You have space (pages)!**
  - **the best related work sections provide additional understanding of the field as a whole**

Related Work: Before or After?

- Pros of discussing related work at the beginning:
  - Give fuller context of your work
  - Answer the questions of knowledgeable readers
  - Give better understanding of the novelty of the work

- Pros of discussing related work at the end:
  - readers now know your work and can more easily understand the differences with existing work

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- ➤ **Approach**
- Implementation
- Evaluation
- Conclusions and Future Work
- Bibliography
- Appendices

Approach

- **Big picture** driving details to follow

- What is the big idea of your solution?
  - Design?
  - Experimental approach?
  - Theoretical approach?
  - New domain?
  - What makes it different from previous?

- Techniques
  - pictures, diagrams
  - idealized example

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- Approach
- ➤ **Implementation**
- Evaluation
- Conclusions and Future Work
- Bibliography
- Appendices

Implementation

- Give details important to
  - achieving your goals
  - proving your claims

- Can someone reproduce your results from your description?

- Why as well as how

Implementation: *Advice*

- Try partitioning the implementation section into modular subsections
  - good writing is sometimes like good code
  - modularize
  - highlight the key ideas

- This is not a diary!

  – stream-of-consciousness writing does not highlight key ideas

- However, sometimes failed or discarded attempts worth mentioning
  - Do so when provides insight for the "why"
  - Example: "I chose clustering algorithm A among several tested because algorithm B turned out to be too slow, algorithm C didn't work in this case, …"

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- Approach
- Implementation
- ➢ **Evaluation**
- Conclusions and Future Work
- Bibliography
- Appendices

Evaluation

- How successful is your project?

- What are the criteria for success?
  - You should state the broad criteria earlier
  - But you can be more precise here

- Experiments to show success?
  - Performance evaluation
  - Quantified user studies
  - Comparison to other methods or products

- Quantitative measures of success
  - Statistical significance of results
  - Comparison to "gold standards"

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- Approach
- Implementation
- Evaluation

➢ **Conclusions and Future Work**
- Bibliography
- Appendices

Conclusions and Future Work

- Can be one or two sections

- Summary of important contributions

- Discuss how you would go forward

- Discuss how others can go forward

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- Approach
- Implementation
- Evaluation
- Conclusions and Future Work
- **Bibliography**
- Appendices

Bibliography

- All papers, videos, … you mention in the text

- All tools you use (and so mention in the text)

- Other references you may have used but not cited in the text
  - e.g., background reading

- Relevant private communications
  - e.g., researcher sends you unpublished performance numbers that you use in text

  "Joe Smith, private communication, 2016"

Bibliographic Form

- Many acceptable forms
  - Different publishers, different forms

- Authors, paper title, publication title, publisher, date, pages. (online pointer)

[8] Jon Kleinberg. "Authoritative Sources in a Hyperlinked Environment." In Proc. 9th ACM- SIAM Symposium on Discrete Algorithms, pp. 668—677. New York: ACM Press, 1998.

Citations in Text

- Citations in text

- Use number in brackets to refer to a bibliographic entry

  "The HITS algorithm [8] also computes a link-based …"

- Using a bibliographic tool makes things easier – e.g., bibtex for latex

- Footnotes for asides – *sparingly*
  "… traverse index in reverse chronological order[2]…"

  _____

  "[2] Although this is not an absolute requirement …"

Citations in Text

- *Do not use a citation as a noun.*

  - You will see lots of published authors do this. It is bad style.

- You have used a citation correctly when it can be removed from the sentence and it is still a sentence.

  - good: "The HITS algorithm [8] computes ..."

    "The HITS algorithm computes ..." is still a sentence

  - bad: "[8] defines the HITS algorithm to compute ..."

    "defines the HITS algorithm to compute ..." is not a sentence

Enhanced Outline

- Abstract
- Introduction
- Problem Background and Related Work
- Approach
- Implementation
- Evaluation
- Conclusions and Future Work
- Bibliography
- ➢ **Appendices**

Appendices

- Optional

- Don't expect readers to look at them

- Uses
  - Data tables summarized in paper
  - Details of long proof, a detailed example, code of a key algorithm
  - Other details important to deep readers

- A luxury of a thesis or "mini thesis"

What Everyone Wants to Know: How many pages?

▪ Averages for a small sample of A-level papers

Section	1 sem. project	thesis
Introduction	avg. 1.5	2-8, avg. 3.5
Related Work	avg. 4.5	avg. 7.5
Approach	1–8, avg.3	1-8, avg. 3.5
Implementation	avg. 10	avg. 13.5
Evaluation	avg. 5.5	avg. 11.5
Conclusions	avg. 1.25	1-7, avg. 3.5

Look at Some Examples

https://www.cs.princeton.edu/ugrad/independent-work/guidelines-and-useful-information#Example_Single-Semester_Projects_from_Previous_Years

https://dataspace.princeton.edu/handle/88435/dsp01mp48sc83w/simple-search?query=2020

Writing Advice – *High Level*

- Write for a general technical audience
  - e.g., all your COS classmates
  - not for your adviser!

- Don't blur your contributions with those of others
  - "We know that …" Your result? Someone else's?

- Get feedback on drafts
  - classmates, parents, …

Writing Advice – *Approach*

- Good writers put themselves in the place of the reader
  - what does my reader know so far?
  - am I saying something my reader can't understand given what they know so far?
  - what do they need to know next?

- Eliminate redundancy
  - "What is the information content of this word or sentence?"
  - remove redundant words, phrases, sentences, paragraphs

- Define technical terms and notation clearly
  - define terms before they are used!
  - write out domain-specific abbreviations first time used,
    e.g., "The Domain Name System (DNS) becomes a bottleneck."

Writing Advice – *Details*

- Avoid unnecessary complexity and jargon
  - From George Orwell's essay on *Politics and the English language*
    - Never use a metaphor, simile, or other figure of speech which you are used to seeing in print.
    - Never use a long word where a short one will do.
    - If it is possible to cut a word out, always cut it out.
    - Never use the passive where you can use the active.
    - Never use a foreign phrase, a scientific word, or a jargon word if you can think of an everyday English equivalent.
    - Break any of these rules sooner than say anything outright barbarous.
- Proofread! Spell-check!

Writing Advice – *Adding Graphics*

- Use figures to help clarify
  - data interpretation
  - architectures
  - interfaces

- Do not overuse figures
  - what does a reader gain by seeing this figure?

- Figure sizes
  - large enough to easily read
  - don't pad paper with unnecessarily large figures

Writing Advice – *Form*

- 12pt Times-Roman font

- 1-inch margins

- double-spaced

- Latex template files posted

Writing Advice – *Procedure*

- Start with an extended outline

- Don't try to write it all at once

- Write something, even a few lines, every day

- "Don't fall in love with your prose: writing and rewriting is what every author does to create papers that are both convincing and clear"

  – [Dr. Rob Fish]

- Use headers and sub-headers
  - helps illuminate logical flow of paper

Summary

- Follow outline but don't be shackled by it

- Don't lose the big picture for the details

- Will this be clear to others?

- Give yourself time to develop the paper day by day

- See posted examples, linked at iw.cs.princeton.edu

Acknowledgements and Reference

- Thanks to IW Coordinator Prof. Rob Fish and Prof. Dave Walker for contributions and comments

- Reference

  Simon Peyton Jones of Microsoft Research Cambridge, "How to write a great research paper"

  https://www.microsoft.com/en-us/research/academic-program/write-great-research-paper/