How To Do Scientific Research In Computer Sciences

(Thorsten Strufe, 2017)

Einführende Anmerkungen

1. Scientific work generally is the art of coming up with hypotheses about the world/ systems/solutions and finding evidence to support them (at best: proofs; still ok: disproving the antithesis ; still ok: measurements or simulation results that support the hypotheses)
2. There are two types of scientific studies (that come to my mind immediately): explorative studies and constructive work.

- Explorative studies analyze a phenomenon observed in reality or on a given system, as witnessed through evidence (you usually will have to actually come up with a solution to collect and interpret the evidence), and come up with explanations for them. They will usually comprise of several smaller steps that are done to gradually explore "the reality"(tm), or more specifically, the characteristics of systems; subsequently trying to provide mathematical models to reproduce and predict them, where reproduction and prediction is validated against additional measurements; and finally explanations why the system produces them, identifying dominating parameters.
- Constructive studies will be by far the majority at our chair. Their hypotheses do not only consist of models that explain systems, and rules that explain, why the system performs as measured, but additionally the main hypothesis how a given problem can be solved (for the first time, or better than the state of the art).
    - In case of security the situation can be somewhat different, there usually are attack or solution papers. Solution papers commonly describe a threat, how it potentially can be exploited and provide a solution algorithm|protocol that is then proven to protect from the threat. Generalizing can be possible ("my new concept can also be applied to the problem X (or even the family of problems X)", or "my methodology can also be used for...").
    - Sometimes the attacks themselves are innovative, relevant or interesting enough that they will be the main contribution of a paper - then the paper essentially will describe design, proof of concept and results of the "algorithm" that implements the attack. Here: the broader the impact of the attack (more people/systems affected), the stronger the contribution. Generalizing in attack papers commonly is very difficult, and hence we commonly don't really learn much from them.

How to write recurrent parts in papers

The Abstract

A good abstract (Zusammenfassung) of a thesis (or a paper) should be like a store sign (Aushängeschild) -- and it should NOT be confused with the introduction. Again: the abstract is NOT an introduction!!! (place as many exclamation marks as you see fit, here). It should instead **be short** and **concise**. Most often it will suffice absolutely to have four sentences as follows:

1. Problem statement: What is the problem tackled in **this** paper (one is more than sufficient for each paper!)
2. Relevance: Why is this problem /really/ a problem and relevant?
3. Response: What is our solution to the problem?
4. Confidence: how do we show in this paper, that our solution is good?

That's it! OK, some of these points may take more than one (short!!) sentence, but help the readers to swiftly judge and understand your text by letting them understand **quickly** what your text is about, how it advances science, and why you think it does so.

The Introduction

The introduction serves to set the stage for the reader. Pick the readers up where they are (consider your audience: don't introduce what P2P and it's problems are, when you write for infocom, don't expect the audience to know what "OSN" are, or why they are a privacy threat, if you write for the IEEE Spectrum (a very broad and general magazine...).

As taken from my template for students, a good introduction should comprise of:

1. % Broad Topic (Broad Topic, potentially little broad background)
2. Thema, special problem we're looking at, motivation (possibly more background for our problem (why is it actually hard?))
3. Broad background, general definitions (Topic, some background)
4. Our goal and our claims (what are we solving in this work?) (Our goal, research question, motivation and relevance (Why is it a problem the reader should care about? Why is it hard?))
5. Requirements for our solution (Requirements for a good solution)
6. Which metrics can we use to show the quality/quantity of our solution? (how can a good solution be distinguished from the rest to measure how good a solution is) (pbly rough definition of metrics)(If space missing the related work may be presented in a paragraph here)

7. Summary of our solution (Overview of our solution and first confidence (how do we show that it's good?))
8. Our contributions in this paper
9. Outline of the paper (Reader's digest)

Then...

A short note regarding related work

Related work is not your enemy, but gives you ``the shoulders of giants" you can stand on; Related work /is your friend!. but: STATE HOW THE RELATED WORK RELATES TO YOUR WORK!! (how is it similar, how is it different?)

(besides, be fair and correct: some of the authors might review your paper...

You will usually write a paper following the structure of Intro -> Background -> Related Work -> Your contribution. The logic here is that in the intro/background you describe the assumptions and requirements, and that you relate your related work to these assumptions and requirements, only to show at the end of the related work section what the missing gap is that you are tackling in this paper (sometimes also what other missing gaps are). Do **NOT** relate the related work to your own contribution, in this structure.

Sometimes some weird people (we've been known to do this) describe the related work at the end, between evaluation and conclusion. ONLY IN THIS CASE IS IT OK TO COMPARE THE RELATED WORK TO YOUR OWN CONTRIBUTION (instead of assumptions and requirements)...

Reporting results / the evaluation

How to report about an experiment / your simulation results.

After answering the following 7 questions you have perfectly described your experiment:

1. What question answered by the experiment?
2. How is it performed?
3. What algorithms?
4. What are the expectations?
5. What came out?
6. It has agreed with the expectations?
7. How can one evaluate this, good / bad?

Further Random Stuff

Kleine Hinweise zur Anfertigung wissenschaftlicher Arbeiten

- Auf jeden Fall wärmstens zu empfehlen ist das Buch "The Craft of Research" von Booth, Colomb und Williams! Ein Exemplar steht in der Bibliothek, es kostet aber auch nicht die Welt
- Wissenschaftliche Arbeiten werden im Deutschen passiv, im Englischen aktiv geschrieben ("Um das Problem zu lösen wird xyz angewandt" vs "To solve the problem we applied...")
- **Konjunktiv** hat in einer wissenschaftlichen Arbeit (insbesondere in den Thesen) nichts zu suchen
- Das gleiche gilt für **Füllworte**, auf die sollte ebenfalls verzichtet werden, wo es möglich ist.
- Re Strukturierung: Eine Aussage pro Satz ("rot ist nicht blau", kurz!), die Aussagen zu einem Konzept/Subjekt/Eigenschaft pro Absatz, die Aussagen zu einem zusammenhängenden Gedanken in einem Absatz.
- Die Argumentation eines Textes soll lückenlos sein: zusammenhängende Sätze müssen direkt auseinander folgen (make the test: simply pick a random sentence and its successor, do you understand the second from the first, are they tightly connected, is anything missing in the logical argument?)
- Eine besondere Gefahr stellen schwammige **Adjektive** dar, die Gegenfragen provozieren, wie *viel* ("aha, und *wie* viel?"), *häufig* (dito), *besonders* ("Und was ist daran besonders?", "In welchem Ausmaß?"), *möglichst* ("und was wenn nicht? Ist das nicht sooo schlimm?", "In welchem Ausmaß?"), *natürlich* ("ist jeder doof, dem sich das nicht sofort erschließt?"), u.s.w.
- Die **Abstraktionsniveaus** sind auseinanderzuhalten. Auch wenns schwerfällt haben Implementierungsdetails in einem Entwurf, oder gar in einer Übersicht nichts zu suchen (es sei denn das dreht sich gerade um die Details einer bestimmten Implementierung).
- Nach *jeder* **Überschrift** kommt Text. Keine Staffelung von Überschriften direkt nacheinander
- Auf **Abbildungen** immer einmal im Text verweisen (Wie in Abb. [x] zu sehen) und jede Abbildung wird in ihrem Caption beschrieben, so dass sie grundsätzlich für sich genommen erklärt ist
- **Zitiert** wird durchgängig gleich. Welche Art der Zitierungsweise angewendet wird (**[autor99titel], Name /kürzel/, [NR]**) sei dahingestellt, aber alle Verweise auf die gleich Art! Bei der Verwendung von Bibtex auf sinnvolle Klassen achten (nicht alles @misc, da gehen zu viele Informationen verloren...)
- Bei Zitierung gilt "most significant first" - also prinzipiell wird die Urquelle zitiert, wenn es zu einem Workshop-Paper ein Konferenzpaper gibt das Konferenzpaper, wenn es dazu ein Journal-Paper gibt das Journal-Paper, wenn es dazu ein Buch gibt das

- Archivierte Referenzen gehören in die Literaturübersicht, nicht-archivierte Referenzen (sollte man eh nicht zitieren, tut man aber manchmal) gehören in Fußnoten (z.B. alles im Web, davon ist nichts archiviert)

A very good outline for papers (and theses) in CS [PaperTemplate.tex]

A good outline for a computer science paper (according to Al Bundy)

1. Title
—> ideally the title should state the hypothesis of the paper
2. Abstract
—> state hypothesis and summarise the evidence that supports or refutes it
3. Introduction
—> motivate the contribution!
4. Literature Survey
—> broad and shallow account of the field, rival approaches, drawbacks of each, major outstanding problems
5. Background
—> states previous work in more detail, where this is necessary for understanding
6. Theory
—> underlying theory, definitions, theorems etc.
7. Specification and Implementation
—> requirements and specifications of implementation
8. Evaluation with related
—> work narrow but deep comparison with main rivals
9. Further Work / Conclusion
—> summarise research, discuss significance, restate hypothesis and the evidence for and against it, - recapitulate original motivation, reassess the state of the field in the light of this new contribution
10. Appendices

Further sources:

- Intro to seminars -- pdf
- http://homepages.inf.ed.ac.uk/bundy/how-tos/writingGuide.html
- http://www-net.cs.umass.edu/kurose/writing/
- http://www.cs.columbia.edu/~hgs/etc/writing-style.html
- Read ``Zen - or the art of motorcycle maintenance" to understand what science and research is (and for the entertainment)

- Read ``The craft of research" to /really/ learn how to conduct research and report about it!
- Some hints on plagiarism: http://www.williamstallings.com/Extras/Writing_Guide.html
- Read strunk & white "elements of style" if you write in English. Re-read it.
- Re-read your favorite parts of "The craft of research"!