Wrapup: Research Papers and Process

Tamara Munzner
Department of Computer Science
University of British Columbia

CPSC 547, Information Visualization
28 November 2017

http://www.cs.ubc.ca/~tmm/courses/547-17F
Today

• writing infovis papers: pitfalls to avoid

• other research pitfalls and process
  – review reading, review writing, conference talks

• final papers and final presentations
  – course paper vs research paper expectations

• reproducible and replicable research

• other course pitch: Rensink
Process & Pitfalls for InfoVis Papers
Idiom pitfalls

• Unjustified Visual Encoding
  – should justify why visual encoding design choices appropriate for problem
  – prerequisite: clear statement of problem and encoding!

• Hammer In Search of Nail
  – should characterize capabilities of new technique if proposed in paper

• Color Cacophony
  – avoid blatant disregard for basic color perception issues
    • huge areas of highly saturated color
    • categorical color coding for 15+ category levels
    • red/green without luminance differences
    • encoding 3 separate attributes with RGB

• Rainbows Just Like In The Sky
  – avoid hue for ordered attributes, perceptual nonlinearity along rainbow gradient
Later pitfalls: Strategy

• What I Did Over My Summer Vacation
  – don’t focus on effort rather than contribution
  – don’t be too low level, it’s not a manual

• Least Publishable Unit
  – avoid tiny increment beyond (your own) previous work
  – bonus points: new name for old technique

• Dense As Plutonium
  – don’t cram in so much content that can’t explain why/what/how
    • fails reproducibility test

• Bad Slice and Dice
  – two papers split up wrong
  – neither is standalone, yet both repeat
Later pitfalls: Tactics

• Stealth Contributions
  – don’t leave them implicit, it’s your job to tell reader explicitly!
  – consider carefully, often different from original project goals
Contributions in research papers

• what are your research contributions?
  – what can we do that wasn’t possible before?
  – how can we do something better than before?
  – what do we know that was unknown or unclear before?
• determines everything
  – from high-level message to which details worth including
• often not obvious
  – diverged from original goals, in retrospect
• state them explicitly and clearly in the introduction
  – don’t hope reviewer or reader will fill them in for you
  – don’t leave unsaid should be obvious after close reading of previous work
  – goal is clarity, not overselling (limitations typically later, in discussion section)
Later pitfalls: Tactics

• Stealth Contributions
  – don’t leave them implicit, it’s your job to tell reader explicitly!
  – consider carefully, often different from original project goals

• I Am So Unique
  – don’t ignore previous work
  – both on similar problems and with similar solutions

• Enumeration Without Justification
  – “X did Y” not enough
  – must say why previous work doesn’t solve your problem
  – what limitations of their does your approach fix?

• I Am Utterly Perfect
  – no you’re not; discussion of limitations makes paper stronger!
Later pitfalls: Results

• Unfettered By Time
  – choose level of detail for performance numbers
  – detailed graphs for technique papers, high-level for design & eval papers

• Straw Man Comparison
  – compare appropriately against state-of-the-art algorithms
  – head-to-head hardware is best (re-run benchmarks yourself, all on same machine)

• Tiny Toy Datasets
  – compare against state-of-the-art dataset sizes for technique (small ok for eval)

• But My Friends Liked It
  – asking labmates not convincing if target audience is domain experts

• Unjustified Tasks
  – use ecologically valid user study tasks: convincing abstraction of real-world use
Final pitfalls: Style

• Deadly Detail Dump
  – explain how only after what and why; provide high-level framing before low-level detail

• Story-Free Captions
  – optimize for flip-through-pictures skimming

• My Picture Speaks For Itself
  – explicitly walk them through images with discussion

• Grammar Is Optional
  – good low-level flow is necessary (but not sufficient), native speaker check good if ESL

• Mistakes Were Made
  – don’t use passive voice, leaves ambiguity about actor
    • your research contribution or done by others?
Final pitfalls: Style 2

• Jargon Attack
  – avoid where you can, define on first use
    • all acronyms should be defined

• Nonspecific Use Of Large
Final pitfalls: Submission

• Slimy Simultaneous Submission
  – often detected when same reviewer for both
  – instant dual rejection, often multi-conference blacklist

• Resubmit Unchanged
  – respond to previous reviews: often get reviewer overlap, irritated if ignored
Generality

- encoding: visualization specific
- strategy: all research
- tactics: all research
- results: visualization specific
- style: all research, except
  - Story-Free Captions, My Picture Speaks For Itself
Research Process & Pitfalls
Review reading pitfalls

• Reviewers Were Idiots
  – rare: insufficient background to judge worth
  – if reviewer didn’t get your point, many readers won’t
  – your job: rewrite so clearly that nobody can misunderstand

• Reviewers Were Threatened By My Brilliance
  – seldom: unduly harsh since intimately familiar with area

• I Just Know Person X Wrote This Review
  – sometimes true, sometimes false
  – don’t get fixated, try not to take it personally

• It’s The Writing Not The Work
  – sometimes true: bad writing can doom good work (good writing may save borderline)
  – sometimes false: weak work common! reinvent the wheel worse than previous one
Review writing pitfalls

• Uncalibrated Dismay
  – remember you’ve only read the best of the best!
  – most new reviewers are overly harsh

• It’s Been Done, Full Stop
  – you must say who did it in which paper, full citation is best

• You Didn’t Cite Me
  – stop and think whether it’s appropriate
  – be calm, not petulant

• You Didn’t Channel Me
  – don’t compare against paper you would have written
    • review the paper they submitted
Conference talk pitfalls

• Results As Dessert
  – don’t save until the end as a reward for the stalwart!
  – showcase early to motivate

• A Thousand Words, No Pictures
  – aggressively replace words with illustrations
  – most slides should have a picture

• Full Coverage Or Bust
  – cannot fit all details from paper
  – communicate big picture
  – talk as advertising: convince them it’s worth their time to read paper!
Paper writing process suggestions

• pre-paper talk
  – write and give talk first, as if presenting at conference
  – iterate on talk slides to get structure, ordering, arguments right
  – then create paper outline from final draft of slides
    • encourages concise explanations of critical ideas, creation of key diagrams
    • avoids wordsmithing digressions and ratholes
    • easier to cut slides than prose you agonized over

• pre-paper/practice talk feedback session: at least 2-3x talk length
  – global comments, then slide by slide detailed discussion
  – nurture culture of internal critique (build your own critique group if necessary)

• have non-authors read paper before submitting
  – internal review can catch many problems
  – ideally group feedback session as above
Final Papers & Presentations
Final reports

- PDF, use InfoVis templates [http://junctionpublishing.org/vgtc/Tasks/camera_tvCG.html](http://junctionpublishing.org/vgtc/Tasks/camera_tvCG.html)

- no length cap: illustrate freely with screenshots!
  - design study / technique: aim for at least 6-8 pages
  - analysis / survey: aim for at least 15-20 pages

- ok to re-use text from proposal, interim writeup

- encourage looking at my writing correctness and style guidelines
  - [http://www.cs.ubc.ca/~tmm/writing.html](http://www.cs.ubc.ca/~tmm/writing.html)

- strongly encourage looking at previous examples
  - [www.cs.ubc.ca/~tmm/courses/547-17F/projectdesc.html#examp](http://www.cs.ubc.ca/~tmm/courses/547-17F/projectdesc.html#examp)
  - Example Past Projects
  - browse 2015, 2014,… reports
Course requirements vs research paper standards

• research novelty **not** required

• mid-level discussion of implementation **is** required
  – part of my judgement is about how much work you did
  – high level: what toolkits etc did you use
  – medium level: what pre-existing features did you use/adapt
  – low level **not** required: manual of how to use, data structure details

• design justification **is** required
  – (unless analysis/survey project)
  – different in flavour between design study projects and technique projects
  – technique explanation alone is not enough

• publication-level validation **not** required
  – user studies, extensive computational benchmarks, utility to target audience
Report structure: General

• low level: necessary but not sufficient
  – correct grammar/spelling
  – sentence flow

• medium level: order of explanations
  – build up ideas

• high through low level: why/what before how
  – paper level
    • motivation: why should I care
    • overview: what did you do
    • details: how did you do it
  – section level
    • overview then details
  – sometimes subsection or paragraph level
Sample outlines: Design study

• [www.cs.ubc.ca/~tmm/courses/547-17F/projectdesc.html#examp](www.cs.ubc.ca/~tmm/courses/547-17F/projectdesc.html#examp)

• abstract
  – concise summary of your project
  – do not include citations

• introduction
  – give big picture, establish scope, some background material might be appropriate

• related work
  – include both work aimed at similar problems and similar solutions
  – no requirement for research novelty, but still frame how your work relates to it
  – cover both academic and relevant non-academic work
  – you might reorder to have this section later
Sample outlines: Design study II

• data and task abstractions
  – analyze your domain problem according to book framework (what/why)
  – include both domain-language descriptions and abstract versions
  – could split into data vs task, then domain vs abstract - or vice versa!
  – typically data first then task, so that can refer to data abstr within task abstr

• solution
  – describe your solution idiom (visual encoding and interaction)
  – analyze it according to book framework (how)
  – justify your design choices with respect to alternatives
  – if significant algorithm work, discuss algorithm and data structures
Sample outlines: Design study III

• implementation
  – medium-level implementation description
    • specifics of what you wrote vs what existing libraries/toolkits/components do
  – breakdown of who did what work

• results
  – include scenarios of use illustrated with multiple screenshots of your software
    • walk reader through how your interface succeeds (or falls short) of solving intended problem
    • report on evaluation you did (eg deployment to target users, computational benchmarks)
    • screenshots should be png (lossless compression) not jpg (lossy compression)!

• discussion and future work
  – reflect on your approach: strengths, weaknesses, limitations
  – lessons learned: what do you know now that you didn’t when you started?
  – future work: what would you do if you had more time?
Sample outlines: Design study IV

• conclusions
  – summarize what you’ve done
  – different than abstract since reader has seen all the details

• bibliography
  – make sure to use real references for work that’s been published academically
    • not just URL
    • check arxiv papers, many have forward link to final publication venue - use that too!
  – be consistent! most online sources require cleanup including IEEE/ACM DLs
    • do pay attention to my instructions for checking reference consistency
      – http://www.cs.ubc.ca/~tmm/writing.html#refs
Sample outlines: Technique (diffs)

- **Abstract, Introduction (same as above)**
- **Related Work**
  - big focus on similar solutions, some discussion of similar problems (same task/data combo)
- **Data and Task Abstractions**
  - much shorter than the corresponding one for design studies, framing context not core contrib
- **Solution**
  - describing proposed idiom exactly, not justifying its use for particular domain problem
  - as above, analyze in terms of design choices, justify why appropriate vs alternatives
- **Implementation (same as above)**
- **Results**
  - less emphasis on scenarios with particular target users
  - more emphasis on characterizing the breadth of possible uses
  - still definitely include screenshots of the system in action
- **Discussion / Future Work, Conclusions, Bibliography (same as above)**
Sample outlines: Survey (diffs)

• Abstract (same as above)
• Introduction
  – discuss the scope of what you're covering, why it’s interesting/reasonable partition compared to visualization as a whole
• Related Work
  – only previous surveys
    • focus on how your work is similar to or different from them, especially wrt coverage
• Main
  – break up into sections based on your own synthesis of themes of work covered
  – you might want a Background section at the start if domain-focused survey
    • where there’s important vocabulary/ideas to establish before diving into main discussion
  – analyze visualizations proposed in these papers in terms of what/why/how framework
    • include images from papers
• Discussion / Future Work, Conclusions, Bibliography (same as above)
Sample outlines: Other types

• see page for implementation & analysis project types
  – implementation, analysis
    www.cs.ubc.ca/~tmm/courses/547-17F/projectdesc.html#outlines

• interactive explanations: meet with me in advance to discuss
Report marking

• required: at least material I’ve listed
  – you may include more material, you may choose alternate orderings

• probable marking scheme (may change!)
  • design study & technique: 12.5% each for
    – intro, related work, abstractions, solution, implementation, results, discussion, style
    – style: 10% main, 2.5% bibliography
  • survey: intro (10%), relwork (10%), main (60%), style (20%)
  • analysis: intro/domain (8%), abstr (8%), relwork (8%), analysis (52%), methods/tools (8%), discussion (8%), style (8%)

• reminder: project content is 60% of entire project mark
  – report is 25%, presentation is 15%
Code / Video

• required: submit your code
  – so I can see what you’ve done, but I will not post
  – include README file at root with brief roadmap/overview of organization
    • which parts are your code vs libraries
    • how to compile and run
    • I do not necessarily expect your code compiles on my machine

• encouraged but not required
  – submit live demo URL
  – open-source your code (if so, fine to just send me that URL)
  – submit supporting video
    • with or without voiceover
    • very nice to have later; software bitrot makes demos not last forever!
  – can be same or different from what you show in final presentation
Showcase image

• new this year: showcase image for projects page
  – 300x300 image
  – call it showcase.png or showcase.jpg
Logistics

• Assignments: Final Presentations on Canvas
  – upload due Tue Dec 12 6pm

• Assignments: Final Report on Canvas
  – upload due Fri Dec 15 11:59pm
    • required & posted: report, showcase image
    • required but not posted: code including README
    • encouraged: live demo URL, video
Final presentations

• context
  – CS department will be invited, also feel free to invite others
  – refreshments will be served, short breaks every hour (or so)
  – order: alphabetical by last name

• code freeze
  – no additional work on project after presentation deadline
  – additional three days to get it all written down coherently for final report
Final presentations: Tue Dec 15 1-5 FSC 2300A

• length
  – 15 min per team presentations, plus 1-2 min questions, 7 teams
  – 12 min per individual project presentations, plus 1-2 min questions, 2 people

• session structure
  – order alphabetical by first name, as on project page
  – 2 breaks, between each set of 3 presentations
  – in theory end by 4pm, reserve buffer of 1 hour extra since we often run over
  – dept invited, friends welcome, refreshments served

• presentation structure
  – slides required
  – demo or video encouraged
    • if plan is for demo, screenshots and/or video for backup strongly encouraged
      – but do practice, demos eat up time!
  – should be standalone
    • don’t assume audience has read proposal or updates (or remembers your pitch)

• slide upload
  – post your slides by 6pm if using your laptops (best), or by 11am if using mine
  – upload to Canvas Assignments: Final Presentations
Final presentations marking

• last year’s template
  – Intro/Framing:
  – Main:
  – Limitations/Critique/Lessons:
  – Slides:
  – Style:
  – Demo/Video:
  – Timing:
  – Question Handling:
Marking: Course overall

• 50% Project, summative assessment at end
  – 15% Final Presentation
  – 25% Final Report
  – 60% Content
  – (penalty to 20% for missed Milestones, pass/fail)
    • pitch, proposal, peer review 1, peer review 2

• 20% Presentations
  – 75% Content:
    • Summary 50%, Analysis 25%, Critique 25%
  – 25% Delivery:
    • Presentation Style 50%, Slide Quality 50%

• 30% Participation
  – 60% Written Questions
    • 6 weeks, 10% each
  – 40% In-Class Discussion & Group Work (pass/fail)
    • 4 weeks, 10% each

• marking by buckets
  – great 100%
  – good 89%
  – ok 78%
  – poor 67%
  – zero 0%
Come talk!

• encourage meeting with me to get advice/feedback before final present
  – chance to get feedback while you can still act on it
  – optional, not mandatory
  – do send email to schedule, can’t meet with all 10 teams in last few days!
Reproducible and Replicable Research
Reproducible research

• 5: 15 minutes with free tools
• 4: 15 minutes with proprietary tools
• 3: considerable effort
• 2: extreme effort
• 1: cannot seem to be reproduced
• 0: cannot be reproduced

Why bother with reproducibility

• moral high ground
  – for Science!

• enlightened self-interest
  – make your own life easier
  – you’ll be cited more often by academics
  – your work is more likely to be used by industry
Reproducibility: Levels to consider

• paper
  – post it online
  – make sure it stays accessible when you move on to new place

• algorithm
  – well documented in paper itself
  – document further with supplemental materials

• code
  – make available as open source
  – pick right spot on continuum of effort involved, from minimal to massive
    • just put it up warts and all, minimal documentation
    • well documented and tested
    • build a whole community
Reproducibility: Levels to consider, cont.

• data
  – make available
    • technique/algorithm: data used by system
      – tricky issue in visualization: data might not be yours to release!
    • evaluation: user study results
      – ethics approval possible if PII sanitized, typically needs advance planning

• parameters
  – how exactly to regenerate/produce figures, tables
    – example: http://www.cs.utah.edu/~gk/papers/vis03/
View from industry

• Increasing the Impact of Visualization Research panel, VIS 2017
  – Krist Wongsuphasawat, Data Visualization Scientist, Twitter

https://www.slideshare.net/kristw/increasing-the-impact-of-visualization-research
Replication: crisis in psychology, medicine, etc

• early rumblings left me with (ignorable) qualms
  – papers: Is most published research false?, Storks Deliver Babies (p= 0.008), The Earth is spherical (p < 0.05), False-Positive Psychology

• groundswell of change for what methods are considered legitimate
  – out
    • p-hacking / p-value fishing / data dredging
    • Hypothesizing After Results are Known (HARKing)
  – in
    • replication
    • pre-registration

– brouhaha with bimodal responses
  • some people doubling down and defending previous work
  • many willing to repudiate (their own) earlier styles of working
Remarkable introspection on methods

• thoughtful willingness to change standards of field
  – Andrew Gelman’s commentary on the Susan Fiske article
    • http://andrewgelman.com/2016/09/21/what-has-happened-down-here-is-the-winds-have-changed/
  – Simone Vazier’s entire Sometimes I’m Wrong blog
    • http://sometimesimwrong.typepad.com/
    • especially posts on topic Scientific Integrity
  – Joe Simmons Data Colada blog post What I Want Our Field to Prioritize
    • http://datacolada.org/53/
  – Dana Carvey’s brave statement on her previous power pose work
    • http://faculty.haas.berkeley.edu/dana_carney/pdf_My%20position%20on%20power%20poses.pdf
When and how will this storm hit visualization?

• they’re ahead of us
  – they have some paper retractions
    • we don’t (yet) have any retractions for methodological considerations
  – they agonize about difficulty of getting failure-to-replicate papers accepted
    • we hardly ever even try to do such work
  – they are a much older field
    • we’re younger: might our power hierarchies thus be less entrenched??…
  – they are higher profile
    • we don’t have vis research results appear regularly in major newspapers/magazines
  – they have rich fabric of blogs as major drivers of discussion
    • crosscutting traditional power hierarchies
    • we have far fewer active bloggers

• replication crisis will be focus of BELIV 2018 workshop at IEEE VIS
  – evaluation and BEyond - methodoLogIcal approaches for Visualization
  – http://beliv.cs.univie.ac.at/
Terrain of blog critiques

• meta: methods for methodological critique
  – Uri Simonsohn post Menchsplaining: Three Ideas for Civil Criticism
    • http://datacolada.org/52
    • don’t label, describe
    • don’t infer motives
    • reach out: contacting authors whose work you discuss before making things public
      – as a heuristic check on tone, imagine going to dinner with authors and their parents that night

• resonates with my own first foray into blog critique
  • https://tamaramunzner.wordpress.com/2016/01/16/on-the-memorability-debate/
    – tone check advice is spot on
      • I *did* go out to dinner with Stephen Few the night I wrote my blog posts!
        – leading me to pick my tone with suitable care
    – I did not reach out, but now I think it would be wise indeed
Other Courses
Visualization course in Psych

• Ron Rensink course
  Special Topics in Perception: Visual Display Design
• http://www2.psych.ubc.ca/~rensink/courses/psyc579/