The Nitty Gritty of PhD Dirty Work. if you can get it

Jon Crowcroft

http://www.cl.cam.ac.uk/~jac22

2026 attendance question: "Where&When is a large language model useful in your PhD?"

Process - wrong!

- Pick Institute
- Find Advisor
- Find Topic
- {work, writeup, publish}*
- s/paper/chapter/
- Submit thesis
- Defend thesis
- Get Faculty Position

Process - better

- Pick Topic
- Find Advisor
- Change topic - not rare
- Change advisor - not necessarily bad
- Change institute - difficult but not un-heard-of
- Do Research (no prescription for this)
- Write thesis (*thesisometer* no particular help)
- Defend thesis
- Get Faculty Position or industry or retire (*)

If in doubt…

- Do a Masters (MSc/Mphil)
- If still in doubt…
  - Do not do a PhD
- If merely torn between several interesting alternate topics….
  - See which advisors/inst. you like best
  - And follow the money

Funds

- PhD funding comes from several places
  - You
  - EPSRC or equiv. via DTAs (or DTCs)
  - EU projects
  - Industry (you may lose IP, but gain a job later)
  - Univ (CISS, Gates) & College scholarships
  - Your country's benefactors
- Most places require 3 years funds up front
  - They don't want you to have to leave for reasons of poverty
  - Speak to possible institute/advisor *with your idea*
  - As this may make them think of the alternatives

Picking Topic Part b
To do, is To Be

- First off, Don't be Hamlet

  - "to be or not to be, and all that jazz"

- Be Jean Paul Sartre

  - "to do is to be"

- And for landsakes, don't be Frank Sinatra

  - "do be do be doo"

- But not Scooby Doo (or Fred Flintstone)

It is really essential to do some actual work

- Work precedes thought in many cases.
- Simon Peyton-Jones' maxim
  - writing a paper being about writing down your ideas to clarify
- Leslie Lamport's:
  - Writing down your ideas in code is an even better way to bring precision to your ideas

Software works for you

- But working code also does work for you.
- You can use it to get results.
  - Results are really useful since they tell you about things:
    - *reproduce*,
    - *contradict*, or
    - *improve* on other peoples results,
- All fodder for chapters of thesis

Zero Day Knowledge

- Most people start a CS PhD with some idea how to write code, and a vaguer idea how to write a dissertation.
  - Some may have written a paper once or twice.
  - The writing bit is a lot easier when you have a system to describe and results to report.
- This means that from Day Zero, you can actually get on with things…
  - …even when *you don't even know what you are doing*!

Some navel gazing errors

- Decisions not to make
  - Big v. small
  - Bottom up v. top down
  - Gap analysis v. synthesis
- or
  - Just do it…

Socialize your ideas

- Meet people, talk to them as often as possible
- Write papers and give talks at every available opportunity
  - Get feedback, listen to it
  - Meet people again.
- If you feel a bit "shy" about your ideas,
  - Try them on local friends/colleagues first
  - Some phd student dedicated workshops too!

Finding that elusive problem…

- If you are only moderately lucky, the devil will be in the details, and…
- hey presto, you have
  - hypothesis (H0) and
  - a plan (P0), and eventually
  - a dissertation (D0) and…
- …a diploma in thinkology (ThD) (TM Ozco)

- So what about those details…

Impact & luck
&
the essence of all true comedy

- If you're very lucky with timing
  - You might have a lot of impact….
  - Even if you made mistakes (loop freedom, not)
  - Actually, making mistakes gets you more citations:)
- You might do really cool work
  - But vanish without trace
- This is not something you can plan for
  - It won't affect the validity of your PhD
  - So don't worry about it
  - I'll say that again: do **not** pursue impact

You might get gazumped

- Imitation is the sincerest form of flattery
  - Doesn't matter provided its during, not before your work
  - Proves you are right:)
  - Risk of being a fashion victim
- On the other hand, you might be so far ahead of other people…
  - They don't even notice you gazumped them
  - Unless they are patent lawyers looking for prior art:-)

Obscurity ain't necessarily bad,
but prepare to be disappointed

- …that you won't get gazumped
  - You might work on something very outre…
- And you will get a cool thesis
  - But you'll never hear of it again with high probability
- On the other hand, a .0001% performance hike which the annual speed up in PCs does anyway and noone uses is a Bad Idea

Misguided heroes

- You might write your own OS
  - This is probably fun but very very risky
  - If you really know where you want to go next and only if your advisor agrees
  - And you don't believe in sleep
- Brave, or Foolish?
- Better to be a cog in a big machine
  - (see below)

"A PhD in a year" -
the anti-hero

- Several students claim they did it in 1 year...
  - Actually, it is fairly common that the work in the dissertation represents 1 year But the previous 2 years were necessary to get in to the state of mind and skills base to do this.
- Note Bene:- you can talk about all those other things
  - etc in your viva -
  - just leave them out of the dissertation!!!

The gateway character

- Like the geni in the lamp
  - In story writing, a gateway is a person who gives you your task:
  - "now Frodo, will you take this ring…"
  - "Your mission, should you choose to accept…"
  - …"to boldly go…"
- Your supervisor/advisor *may not be the gateway*
  - Often its another student or person at a conference, or author of a paper you dismantle
  - The more you socialize, the more likely you find your elusive gateway

Really useful may or may not be research

- Really useful stuff might not be research in itself
  - but the actual research…
  - …turned up in doing them, of itself
  - i.e. (again), the devil is in the details
  - Research is almost fractal…

Being a cog
in a big machine

- Being part…of a really **big** organisation is good, *not* bad
  - Your name may be lost in a cast of 1000s
  - But you are at least on the credits
  - And that's what your next job is likely to be like too
  - Key in big projects:
    - is to partition work cleanly
    - and define shared components cleanly

Writing it down -
papers may be chapters

- If you are at a loss(pun:-),
  - write it down or
  - say it to someone
- Levels of abstraction are good
  - math v. code v. design tools
  - "lab" books
- Papers may become chapters
  - But not always -
  - only if you are lucky:
  - that progression of work matches thesis *story*
- A thesis is more like a novel than a paper
  - But a chapter like a short story version of a paper
  - Narrative is good.

What next?

- Framing your PhD:-
  - Do you wannabe an academic
  - Or an industry lab researcher
  - Or an industry builder
  - Or do a startup
  - Or a consultant
- roughly even mix in my 60.3 PhDs so far
- Timing of papers is slightly affected by this

Dissertations

- ## Shorter is better
  - Proofread by normal humans is better
  - Spellchecked by programmes is better
  - Error bars, bibliographies, captions, glossaries, legends are all better
  - Giving advisors and friends more than 24 hours warning is better
- ## In defense of examiners….
  - They work quite hard for very little "pay"
  - So make their lives easier
  - Prepare for viva properly

On writing dissertations…

- There's no formula
- Two patterns common
  - Glom papers together and see what you have
    - Edit edit edit
  - Read through all your notes/papers,
    - throw them away and sit down and
    - write from start to finish
- Hybrids abound…
- If in doubt, leave it out.

Examiners/Defenses (UK)

- Examiners are usually 2
  - 1 local (generalist), 1 (expert) external
  - Defense is private and 1-5 hours
  - Some inst. Let advisor be present, but silent
- There's no formula for a UK defense
  - But having 15-20 min summary of your contributions ready does no harm
  - And offering any errata you've found between submission & defense doesn't hurt either.

How examiners are chosen

- Typically, a degree cttee ask advisor for suggestion for external
  - But BOGS filter out for inappropriate (conflict, overworked, inexperienced)
  - Internal is usually from different group
  - Or possibly different but related department
- Some places let you have 2 externals
- Some let you have 3 examiners!
- Europe (mostly) has a huge committee
  - (and public defense as well as sometimes private)

Viva outcomes

- Range from accept, through to
  - minor corrections (very common),
  - major corrections (quite common),
  - re-submit (with or without viva) occasional, or
  - (rarely) MPhil "only" and
  - a big no (very very rare)
- If your friends and advisors all say ok,
  - then you ought to be in one of first 3.
  - There's always some uncertainty
- make sure corrections requested are
  - very well specified (including time frames)
  - (although this is really the local examiner's job)
  - & understood (your job)

Q&A&B

- Questions….?
  - No, Answers…?
- No, ok, so – the attendance question's answer
  - No you don't need to publish, but it is worth a try for several reasons, not least it is a hint to the examiners that the work already passed the bar necessary for recognized/original research, but also as a way to stage your work, get early independent feedback, and maybe job offers☺

Acknowledgements pt 1

- Some students were probably damaged in the making of this work:
    - https://www.cl.cam.ac.uk/~jac22/paststudents.html
    - CL netos list members
    - https://www.cl.cam.ac.uk/~jac22/students-examined.htm

Acknowledgements pt 2

- Some ideas were probably broken in this work:
  - Prefetching&Better TCP, tcp friendly video, cbt=pim, pip=ipv6, reflective ui toolkits, qos cast, iptv carousel scheduling and patching, delay bound ethernet, scale arch for group comms, hetero mmconf, emergent GP, self org transcoder, anything is better than TCP, provision sprint, VOIP call routing, multicast games&metaphors, re-ECN, OO Met Management, Provisioning gMPLS, pain thresholds and mmconf , Complexity of IDPR/IDRP, matchmaking clouds, centralised DNS, ignoring frame checksum errors for media, internet coord systems, hybrid geo/topo MANET, securing mobile OS, go faster IDS sigs, economic complexity of netarch, auto virus containment, opp net coding for mesh, social forwarding for DTN, censorship, redecentralisation, data center transport, etc etc