COMP90044 Research Methods

Qualitative Research

Quantitative Research

- Quantitative research focuses on numeric values
- Typically from experiments or field data
- Which are then tested statistically to test
- …and hopefully validate…a hypothesis

- To achieve this stage in research you need to have a clear understanding of the mechanisms at play
- …and also for those mechanisms to be quantifiable

Qualitative Research

- Qualitative research looks at descriptors and labels
- Typically from interview data, focus groups or other human-language sources
- …which are then examined for patterns of meaning
- …typically to *form* a hypothesis (e.g. a model)
- …of how these world 'works'

- Often there is little formal prior knowledge
- …though there typically is some related work

The Big Picture

- In some areas, there is little qualitative research to do these days – the domains are mature and progress can continue along quantitative lines
- However, in newer areas of science, knowledge is embryonic, and uncertainty about *what* to test quantitatively is high
- Often quantitative research engages in 'fishing exercise' statistical analyses to find potential measures that influence a behaviour (e.g. via data mining)

The Big Picture

- However, that is philosophically inept, and also you may not even have quantitative data available!
- In what are called the 'human sciences', e.g. :
  - (Software-, Construction-) Engineering processes
  - Human-computer interaction
  - Information systems
  - IT and Engineering management
- …you often rely on assessing human thought and experience (e.g. expertise, psychology)
- And the issues are fundamentally on linguistic material (audio, written, etc.)

Qualitative Research

In engaging in Qualitative Research, you're likely to be in one of two situations:

- Building on an existing model or theory of how a process works (e.g. understanding it in a different context, extending it to address a gap)
- Working on uncharted territory, where there is new practice or an absence of a theory on existing work

Other cases do exist, but something like 90% fall into these two cases

Qualitative Research

- Qualitative research is primarily *hypothesis forming*
- We seek to build up a new* theory (e.g. model)
- …from gathered linguistic material
- Sometimes this may be followed by a *hypothesis testing phase*, e.g. in a field study or survey, which is used to validate our model
- I'll emphasise the *forming* stage today, but point to how qualitative hypothesis testing is done
- * (or extend or adapt an existing)

Coding

- Coding is the main task in qualitative work
- Reading text and placing codes on spans of text that represent an idea
- In the most basic format, we do what is called "flat coding" where there is no structure to the codes
- In more complex forms, there is "hierarchical coding" where ideas form a tree or hierarchy of more abstract ideas, e.g. "happy" and "sad" (could) fall under the broader code of "emotion"

Coding

- So where do these codes come from?
- Two fundamental approaches:
- "in vivo" coding: create codes from scratch when reading the raw text, with no prior assumptions
- "a priori" coding: use existing concepts as codes, either from a theory or previous coding scheme
- …for the record, a-priori coding is almost always done in the context of hypothesis testing

Example Case

- Project: investigate how academics in the humanities (history, literature) find information for their research
- Various existing theories on "information seeking", but are there behaviours not seen in the previous literature?
- …Existing theories may not cover all behaviours, as they primarily date from 1988-1998.
- Would you use a-priori or in-vivo coding?

A-Priori Coding

In A-priori coding, codes or labels come from a theory – let's take an example

- Ellis' theory of information seeking describes several behaviours such as…
- Monitoring : looking regularly at a source of information to check for new material

We would use the label of "monitoring" in our coding scheme, e.g. for the following text:

- "I go to the journal website each week to see if any new pre-prints are available"

A-Priori Coding

- Ellis' model is in fact quite simple: 6 or (revised) 8 behaviours, with no hierarchy
- Thus, it forms a natural "flat" coding scheme
- Where theories have hierarchical structures, they likewise will naturally form hierarchical coding schemes
- You should ensure that codes are not ambiguously named, so that they may be confused with each other – usually theories or models avoid this BUT it does happen!

In Vivo Coding

- In-vivo or 'inductive' coding is where we identify themes from the data itself
- We first read over the content, then identify potential themes or codes from the content (considering our research question)
- This is generally an iterative process, as new codes can emerge as we label the text with other codes

Examples

Here are four interview quotes:

- "I'm not sure … well, you have intuitions sometimes and search the British Library or an archive"
- "You can never be certain where the right place to look is… I often get given tips by colleagues…"
- "I generally check each month on the leading journals – you never know what you might find"
- "Other experts quite often point you in the right direction – and new journal articles and archive catalogues, I usually start there"

Completeness

- You want to discover each instance of a code, not just a few to indicate it might be there
- Ideally count the number of instances (more instances suggest greater importance)
- …and the number of people or sources – one might mention it six times, or six people once
- Once a code is complete, and no more examples of it can be found, that code is "exhausted"

Examples - Uncertainty

Here are four interview quotes:

- "I'm not sure … well, you have intuitions sometimes and search the British Library or an archive"
- "You can never be certain where the right place to look is… I often get given tips by colleagues…"
- "I generally check each month on the leading journals – you never know what you might find"
- "Other experts quite often point you in the right direction – and new journal articles and archive catalogues, I usually start there"

Complementary or Related Codes

- In our interviews here, we may be interested in where information comes *from* – the source
- There are a few different types of codes here, e.g.:
  - Journals
  - Colleagues
  - Libraries and Archives
- No doubt others may be found elsewhere
- Notice the third item here is perhaps two concepts – sometimes you end up splitting a label later..or joining if two become closely related

Examples - Sources

Here are four interview quotes:

- "I'm not sure … well, you have intuitions sometimes and search the British Library or an archive"
- "You can never be certain where the right place to look is… I often get given tips by colleagues…"
- "I generally check each month on the leading journals – you never know what you might find"
- "Other experts quite often point you in the right direction – and new journal articles and archive catalogues, I usually start there"

Doing Coding: Tools

- Coloured highlighter on printouts is one traditional method – good for smaller samples
- There are computer-based markup systems such as nVivo and Atlas.ti – good for large-scale work, but quite a learning curve
- In the middle is using (Microsoft) Excel, and using a cell for each quotation with a code…and organising the codes in rows (or columns)
- Whatever method you use, qualitative coding is *time-consuming* – there is no quick hack!

Doing Coding: Theories (Methods)

- There are several different qualitative "schools of thought" or methods… two major ones are:
- "General inductive approach", introduced by David R Thomas (American Journal of Evaluation, Vol. 27, Issue 2, Sage Publ., June 2006, pp. 237-246)
- "Grounded Theory" from Anselm Strauss and Juliet Corbin (Basics Of Qualitative Research, Sage Publications, 1990)

Doing Coding: Theories (Method)

- General inductive approach is "lightweight"
- …fewer concepts, easier to adopt

- Grounded theory is "heavyweight"
- …many different, interconnected concepts
- …can take years to learn fully

- Very many people claim they are doing "grounded theory" when they really are not!

Choosing a Method

- If you are doing a primarily qualitative research approach for a PhD, it may be worth taking up Grounded Theory … it's designed for bigger work
- If doing taught masters, the general inductive approach is FAR more appropriate
- …and it is primarily what I use in my own research

Choosing a (Domain) Theory

- If you're expanding on a theory in your area, then codes from it are probably easily extracted
- However, if new to an area, or choosing a theory, then you might need to pick from a number
- First consider the theories from principle (i.e. what is your research question, etc.)
- …after that if there are multiple options, consider which would have more complex or simple codes
- …drop all complex or very simple ones

Choosing a (Domain) Theory

- Ideally consider your choices before designing your data capture, as the theories (and hence codes) you *might* use may well influence your interview questions etc.
- Then refine and confirm your choice of theory after you see your data – sometimes this reveals more fully *which* theory works
- …and even better, pilot out your design of both interview and coding first
- …then choose before any main data-gathering

Reporting Coding

- Avoid being vague in your reporting
- While qualitative work is not dominated by numbers, coding counts are a major part of your argument
- They show *exhaustivity* or *completeness*
- …and precision in your work generally

- Also use quotations to drive home your points

Reporting Coding

Bad:

- "Many interviewees expressed uncertainty about how they found information"
- …how many is "many" – a lot? – a majority?

- Better:
- "More than half (13/20) interviewees expressed uncertainty about how they found information"

Reporting Coding

- Even better?
- "Nearly 2/3rd (13/20) interviewees expressed uncertainty about how they found information. However, this was only noted in passing, with most only making the observation once (15 instances across 13 respondents)".
- …here we are using code instances to indicate how many times people returned to the same theme
- This can get fussy if overdone, but useful if you want to emphasise how important or not a code is

In Vivo Coding: Code Exhaustion

- In-vivo coding is iterative – more codes emerge as you progress.
- You list these as they emerge
- …then exhaustively code each one.

- At some point, there are no remaining codes to mark-up the content with
- Then you read over all your content again to check for no new codes

In Vivo Coding: Code Exhaustion

- When you can find no new codes, it is said that you have coded to exhaustion
- This is the ideal point to reach:
- All codes have been identified
- …and all codes applied to the whole text
- There is no new coding to be done

Reliability

- The problem with both creating codes *and applying them* is consistency and reliability
- Doing it just yourself is…hardly bullet-proof
- …you came up with the codes and their application

- Best to do is to do some inter-coder testing
- …another person uses your codes on sample text
- …you then assess how consistent your coding is
- There is a statistical family of tests (kappa tests) to check inter-coder reliability

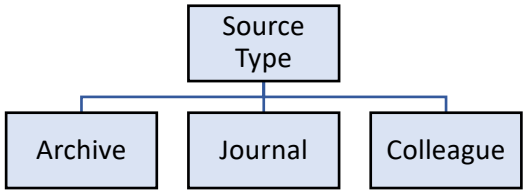
Inter-Coder Reliability

- What one usually does do is start with a sample
- Then reiterate the coding by all coders (2 minimum, 3 better, 4 quite hard!)
- …until the reliability is increased
- Then do the main coding and periodically test for agreement
- This retains a high-level of intercoder reliability (what is "good" varies from field to field)

Code Quality

- The codes themselves are an issue
- Do different people come up with similar codes?
- Are the codes overlapping or clear?
- … some points can be addressed by inter-coder reliability, but this doesn't check the codes directly
- We can verify codes in part by trying multiple schemes by different coders, and *converging* on a single scheme
- …or turn to theories in the field to see if codes align with existing concepts or terminology

Hierarchical Coding



Hierarchical Coding

- Supported in nVivo etc. automatically – as long as you tell the system it's hierarchical, things are done for you
- For manual methods, you need to aggregate any analysis yourself (obviously)
- Usually you report the data as follows:
- "We abstracted five high-level codes, with seventeen low-level codes in total"

Reporting Codes Themselves

You will report:

- How many codes you arrived at
- Indicate their relative prevalence (occurrence)
- Their structure, if hierarchical
- Parts of your coding that are not used in your paper, you will generally not report…except perhaps in passing
- "From Ellis' model we commenced with 8 a-priori codes for each of the behaviours reported in [Ref] and in-vivo arrived at a further thirty six codes, which were grouped under seven higher-level codes. In this paper, two of the seven high-level codes and their children are not reported, as they fall outside this analysis"

The "lightweight" approach

- Sometimes with simple exit interview data from an experiment, all the above is way too much work
- Rather, focus on "selective coding" – find your major codes, code *only* them
- Also don't produce *exhaustive transcripts* or coding, but rather abstract only the most pertinent quotations, and simply count the number of occurrences
- This isn't *good* practice in qualitative research, but is fine if the *main* contribution is quantitative

The “lightweight” approach

This approach is also appropriate in very short material, which is increasingly commonly used in research:

- Tweets and other social media
- Annotations of user actions when observing a task, where you are marking up things such as “the user backtracks” or “engineer corrects settings”.
- Often also video analysis ends up this way unless it is complex (it seldom is in engineering or CS)

Overview

- In qualitative research we are usually either:
- Forming an understanding of a poorly-known area
- Extending existing theory to a new (sub-)domain
- Mark-up or coding is labour intensive, and can be helped by digital tools (Atlas.ti, nVivo)
- …and we can adopt different theoretical frameworks
- Lightweight approaches better in small data sets