Unit 7: Data Annotation

Building a typology

Lesson 49

Derek Ruths

Overview of unit

Objectives:

- Understand how to approach data annotation (whether automated or manual)
- Know how to run a small human annotation task

1. Typologies
2. Building a typology
3. Applying a typology – human coding
4. Confirming an annotation
5. Applying a typology – classifier

Lesson overview

Objectives

- Know the steps involved in building a typology
- Strategies for developing solid category definitions

Outline

- Process for building a typology
- Open coding
- Ensuring typology properties

Typology design objective

A document consisting of…

- Motivation & context – why this typology needs to exist
- Overview of the types and their relation to one another

overview

- List of types. For each type:
  - Concise definition
  - Positive examples with inclusion rationale
  - Negative examples with exclusion rationale
  - Edge cases with inclusion/exclusion rationale
- Argument or evidence for comprehensiveness

annotation guide

thing that look like they might fit

precise

like a page for each type

look like they could go either way but we decide

Building a typology

A comprehensive, sharply-defined categorization system

1. Get representative data always!

2. Get typology (find existing if it exists, or build your own using open coding)

3. Sanity check: evaluate typologies on representative data … can YOU make them work?

3b write your guide

4. Human test: Evaluate typologies on representative data with "expert" coders… people you trust and believe can apply the typology as defined.

5. Does typology work? If yes, done! If no, adjust the typology and go to step 3.

Do we agree w/ what the coders said

Do the coders all agree with each other

train your coders with your document

Developing a typology through open coding → exploratory

1) • Take a sample of data

2) • Go through the sample and come up with categories → do it yourself, as a human

3) • Review categories – are there any that are…
  - Related or overlapping? Should these be merged?
  - How "solid" is each? Assess whether it's a real thing… could these fit in another category? Do we need this level of resolution?
  - Are there any gaps (kinds of things that could happen, but you haven't seen?) … go find some examples of these if you can.

eg feel good ← valid but not super solid

eg golf 2
tennis 1
sports 18

} should fold into sports

is it objective-ish? is it well defined?

ask is my data accurate?

Open coding example...

Tweet	Type of weather
It's pouring outside.	
Just came inside soaking wet.	
Blizzard conditions out there! #hotchocolatetime	
Going to get wet catching the bus today!	← maybe need precipitation category
Sunglasses weather. Can't wait to take a walk.	
Pouring myself some cereal this morning.	

try until you arrive at a set of categories you feel comfortable with

↳ then test by writing concise definitions

Ensuring a typology is comprehensive

- Gather and look at extensive sample – if typology applies everywhere, chances are good it is near comprehensive.

- Catch-all category "other" – worst option

→ can be tempting...

→ we could have one but it needs to be very critical, otherwise it's an easy out for human coders and confusing for machines

dont make it a garbage can (but could have trash, recycling, compost)

Ensuring a typology is well-defined

- Each definition should have rules for when they apply (and don't)

- Each definition should be discernable from the data

- Make sure there is a (not too broad) way of handling ambiguous data.

Ensuring a typology is objective-ish

- Some types may be inherently subjective (beauty, goodness, acceptableness, etc…)

- Truly subjective categories are rarely useful – they will vary based on who you ask!

- To avoid subjective types, ground the definition in a point-of-view
  - E.g., verifiability
  - E.g., edible

Key realities

Building a typology requires looking at LOTS of data

Building a typology requires being comprehensive

Building a typology requires an iterative (potentially long) process

Lesson wrap-up

Takeaways

- Building typologies takes time and patience
- Building typologies involves looking at a lot of data

Up next

- Manual annotation (human coding)