Best practices that could help avoiding the mess

Volker Steinbiss

RWTH Aachen University / Accipio Consulting

steinbiss@informatik.rwth-aachen.de

ACCIPIO CONSULTING

RWTH AACHEN UNIVERSITY

My world

- from mathematics to engineering / physics style
  - build a machine that works
  - test a model on data
- data driven approach
  - knowledge comes from data
  - solution is evaluated on data
  - mathematical („statistical“) formulation of the recognition problem
    - translation into another science, but still challenging, interesting, inspiring!
    - mathematics to model lack of understanding
  - methods scale well (more data lead to improvement)

Avoiding „the mess“

- Research in automatic speech recognition requires corpora of acoustic recordings
  - typically really large (10s/100s of hours)
  - need to be collected, transcribed and validated
  - often exchanged with research partners
- If you scale up, it is a different world
- Some people seem to drown in problems, others to manage it easily

- This presentation: observations / best practices that have been helpful

“The mess” – Examples

- You run the same software on the (same?) data and cannot reproduce the results.
- Your script cannot process all files due to assumptions on file names.
- A Windows/Mac/Unix filename looks strange on Unix/Windows/Mac.
- A text does not display well due to character encoding.
- The transcribers took the guidelines verbatim (and not in the sense you intended) and the work of weeks is wasted.

Size matters

- Make sure that everything
  - scales up nicely
  - runs automatic
  - is reproducible

Formats

- file names
- different files have different file names
- tough like a parser - no discussions
- must support automatic processing

Character encoding

- Migrate and stick to it: UTF8
- All our tools are UTF8 compatible
- Transcriber migrated to UTF-8 encoding
- NIST toolkit does not support UTF-8 yet, so need to convert

- UTF8

- Nice if you work with different languages! Great for Arabic and Chinese.

No assumptions

- The problem is about the things we think we know,
- the communication that we think we understood,
- the things that go without saying (as they are so self-evident!)
- Examples: New languages bring surprises with them
  - How many words are in “qu’est-ce que c’est”?
- Phone numbers
  - phone numbers consist of 7 digits – sure!
  - Telefonnummern können unterschiedlich lang sein – klar!

Concepts

- Name it
- Clarify
- Invent nomenclature
- Qu'est-ce que c'est?

- give it a name
- invent nomenclature if necessary and use it consistently
  - EPPS: final text editions, verbatim, ASR output
- Example from speech recognition: what is a word?
  - qu'est-ce qu c'est?
  - different ways to make words from this. My take on it:
  - consistent, automatic
  - good (for ASR, longer units)

Raw data

- don't throw away information
- wrong choices are costly
- automatic checks
- understand the legal situation, but you might choose to *not*
  - ask for permission
  - draw in lawyers

- There is an abundance of data - start collecting raw data early
- Example European Parliamentary Plenary Sessions (EPPS)
  - we collect satellite recordings since 2004
  - currently 15 languages + original
  - running effort 8h/w
  - transport stream, unpack, cut, ...
  - find approximate transcriptions on the web
  - TC-STAR Projekt
  - for English there is ELRA transkriptions and audio

Guidelines

- capitalization: recommend true casing (such as it would be within the sentence)
- use spell checker
- Acronyms: no gold standard
- Language dependent considerations: in English, contraction – "we'll" – how to transcribe this?

- Guidelines cover e. g.:
  - Capitalization: True casing
  - Acronyms and spelled acronyms:
    E.g. USA:
    - U_S_A?
    - U. S. A. ? U S A ?
    - ... Important: Define it.
- Language dependent considerations: How to transcribe “we'll” ?
- Example: steinbiss@informatik.rwth-aachen.de is transcribed steinbiss at informatik dot R_W_T_H hyphen aachen dot D_E

Errors occur

- Corrections → implies you need versioning
- Document the input in the output
- Make sure you can reproduce results
- Make statistics and checks
- Make a backup
- Operating systems and text files
- Use XML

- If an error occurs, you want to know.
- Corrections → versioning
- Statistics and checks
- Do not make assumptions (like upper limits) ... but if so
  - communicate them
  - make automatic checks

Humans

- Helpful: same coffee machine
- Best: people who are aware of the challenges

Is there more about it?

- Comments or questions?

  Added after presentation:
  RWTH tools on
  http://www-i6.informatik.rwth-aachen.de/web/Software/index.html

- Thank you for your attention!