

**Goal** ⇒ What are the requirements for a system to perform well on the long tail disambiguation task(s)?

- **Questions for discussion (30 mins):**

- How should a system be aware of the long tail in each phase (training, development, test)?
- How do systems perform WRT the long tail? What can we learn from it?
- What should a system have in order to perform well on the short and long tail WRT:
  - Training data that is richer WRT the long tail (features, model, knowledge base, corpus)
  - Development as close as possible to the test (what is the link to domain engineering?)
- How to incentivise systems to focus more on the long tail?
- How to build systems that can easily switch between contexts of time, topic, location, community?
- How to detect the long tail details in the task?
- How to exploit/get knowledge about the context: time, topic, location, community?
- Is there one system for all tasks or different systems for each task? If the latter, what system combines the output of different systems?
- Do we need different systems every 5 years? Or do the system need to adapt themselves while the world changes?
- Instead of overfitting to data popularity you want to fit to the task popularity.
- Do we need an overarching AI task (e.g Watson for Q&A) that involves reasoning and a goal?

- **Hands-on data (90 mins):**

- System error analysis
  - Option 1: the participants take their own system and task, and analyze the long tail cases
  - Option 2: we prepare two system outputs on the same data for WSD
- In any option, report on the performance, compare to at least another system, and discuss the outcome with respect to the long tail.
- Alternative: analyze the full NewsReader output or Babelfy output (<http://babelfy.org/>) or other system of your desire
  - We will prepare a number of ready NewsReader-processed files
  - We provide the participants with a link to the NewsReader demo API (<http://ixa2.si.ehu.es/nrdemo/demo.php>)

- **Concluding the session+recommendations (30 mins):**

- How can we force systems to avoid shallow solutions to the long tail data in the SemEval 2018 task?
- What are the tasks and how are they interconnected?
- How do existing systems perform on the long tail?

- How to build a system that performs well on the long tail?
  - How to build sustainable systems that also perform well in the next 5/10 years?
- **Preparing presentations (15 mins)**