

# Main Semester-Long Project

Teams of three students

Combinations across sections are OK

- ▶ Problem
  - ▶ Interesting to you
  - ▶ Some prospect of being useful broadly
  - ▶ Needn't be new but may have a new twist on an old problem
- ▶ Envisioned software artifacts
- ▶ Underlying sources of knowledge
  - ▶ Lexicons or location services
  - ▶ Software libraries
  - ▶ Datasets for evaluation
  - ▶ Way to have humans help evaluate
- ▶ What you will contribute to the world's body of knowledge?

# Software Artifact

Not needed in detail in the early report

- ▶ High-level view of your method
  - ▶ Identify what knowledge is needed to carry out the method
  - ▶ Identify what knowledge is available
- ▶ Processing “pipeline” or graph
  - ▶ Main components, ideally mostly based on existing libraries
  - ▶ What *you* will *add* to complete the artifact
- ▶ Don't get stuck in product-like details

# Scientific Thinking

Critical thinking going beyond the artifact

- ▶ What do we learn from the exercise?
- ▶ What reusable knowledge will you create?
- ▶ State hypotheses that relate to the main topic
  - ▶ One or more about the quality of your solution
  - ▶ One or more about the effectiveness of specific components in your approach
- ▶ Describe how you will evaluate these hypotheses

# Hypotheses and Evaluation

The nature of the evaluation depends on the specific hypothesis being evaluated

Hypothesis  $\neq$  assumption

- ▶ Should be interesting in that an answer would affect how future developers would solve their problems
- ▶ Should nontrivial and nonobvious
- ▶ A comparative framing helps
  - ▶ Vary the nature and amount of input (data or supervision)
  - ▶ Vary the methods
- ▶ Good to identify one or more baselines ( $\approx$  prior methods)
- ▶ In typical artifacts, multiple components (or capabilities) make it difficult to figure out the relative importance
  - ▶ Ablation studies: consider the components and capabilities separately

# Immediate Actions for Students

- ▶ Look ahead in the course schedule
  - ▶ You don't need to read up in detail
  - ▶ Try to get a sense of what topics are relevant to this course
- ▶ Look at what kinds of research are showing up
  - ▶ Google Scholar may be easiest to search
- ▶ Identify some themes that you find interesting
- ▶ Identify project partners
  - ▶ Use the themes to find like-minded people
  - ▶ Discuss working styles and schedules for compatibility
  - ▶ Discuss how hard they and you will work on the project
- ▶

# Project Topic Ideas for NLP

Extremely small sample: just for discussion

- ▶ Understanding sender's intent from email
  - ▶ Extracting to-do items
  - ▶ Extracting which tasks are assigned and which are completed
- ▶ Developing a chatbot for car repair
- ▶ Dealing with custom named entities, e.g., apps, equipment
- ▶ Resolving pronouns <https://paperswithcode.com/paper/bridging-anaphora-resolution-as-question-1>
- ▶ Dealing with novel words  
<https://arxiv.org/pdf/1811.03866.pdf>