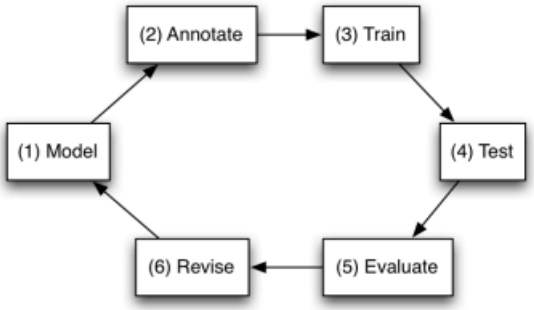
Annotation for Machine Learning

Annotation = The action of annotating a text or diagram

Layers of linguistic description

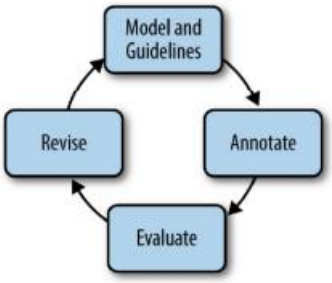
- **Syntax:** how words and phrases are combined into phrases and sentences respectively
- **Semantics:** meanings of phrases/sentences and relations over these meanings
- **Morphology:** smallest units of language that has meaning or function (aka morphemes)
- **Phonology:** sound patterns of a particular language
- **Phonetics:** sounds of human speech, and how they are made and perceived
- **Lexicon:** words and phrases used in a language (vocabulary)
- **Discourse analysis:** exchanges of information and the flow of information across sentence boundaries
- **Pragmatics:** how the context of text affects the meaning of an expression and how to recover a hidden/presupposed meaning
- **Text structure analysis:** how narratives and other textual styles are constructed to make larger textual compositions

MATTER cycle



- **Model** the phenomenon
- **Annotate** with the specification
    - **Design specification**
        - Consuming tags
        - Non-consuming tags
    - **Write guidelines**
        - Span of the tag
    - **Annotate**
    - **Evaluate & revise** (if needed)
        - Understanding or just agreement?
    - **Adjudication & gold standard**
- **Train** algorithms
- **Evaluate** the results
- **Revise** the model and algorithms

MAMA cycle



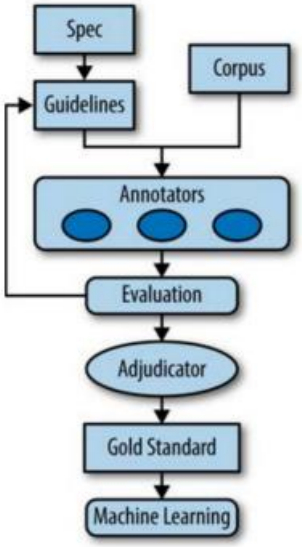
- **Model**
- **Annotate**
- **Model**
- **Annotate**

Generality vs Specificity

The model is captured by a specification, or spec. The spec is the concrete representation of your model. So, whereas the model is an abstract idea of what information you want your annotation to capture, and the interpretation of that information, the spec turns those abstract ideas into tags and attributes that will be applied to your corpus.

Why use a specification:

1. Scale
2. Generality



What do we have to know about generality and the difference between generality and specificity here????

XML vs JSON vs JSONL

Why XML:

- Limitless customisation
- Inline annotations are more interpretable
- Typeless
- Supports namespaces
- More secure

Why JSON:

- Easy (familiar with minimal dependencies)
- Fast
- Free
- No mapping

Why JSONL:

- Appendable
- Adaptable

Inter Annotator Agreement (IAA)

Accuracy and F1 score don't take expected chance agreements into account.

Measures which do take expected chance agreement into account:

- **Cohen's kappa:** two annotators annotating each subject with a category
  - Works only if there are two annotators
  - Annotators have to annotate the exact same items
- **Fleiss' kappa:** each subject was annotated *n* times with a category
  - Works with multiple annotators
  - Annotators can be annotating different items