Day 9: NLP Semantics

- What is a good representation for information (in stories, books, sentences, etc.)?
  - It'd be nice to be able to reason about it
  - Long thought to be logic-based, but that concept has come out of fashion
- AI isn't intelligent if it doesn't actually understand what's going on
- Limits of Syntactic Parsing
  - You get a parse tree, but what do you do with that?
  - Ambiguity of meaning:
    - Same sequence of words means multiple things
    - Non-gramatical sentences that make sense in context
    - Anaphora & Reference: pronouns, "Sarah shot Sally. She was dead and she felt guilty", etc.
    - Inferences need to be made to understand what's happening, ex: `The balloon went into a tree and burst. The baby cried. Mary glared at John and picked up the baby.`
      - The balloon was filled with helium, and the baby liked it. The tree popped the balloon, which made the baby unhappy. John was responsible for holding the balloon. Mary picked up the baby to soothe it. Implicitly, John and Mary are probably married and the baby is probably their child, etc.
    - Multiple parse trees for the same sentence
    - Non-literal language
      - Idioms, Metaphors, Analogies, etc.
    - Language is constantly changing, and you need to keep up to date
    - Sentences that are weirdly worded, but intonation makes them easy for humans (ex. `The horse [which was] raced past the barn fell`)

- How to internally represent meaning?
  - Canonically (ideally across languages)
  - Easy to do inferences on, etc.
  - One example: Case filler structures: `event: meal; actor: Mary; location: unknown; object: spaghetti; instrument: chopsticks;`
- Semantic Grammars
  - Parsing grammars aimed at parsing actors, etc. instead of grammatical structure.
  - Often executed as context-free languages or ATNs
    - Can be thought of like a finite state machine (nodes, etc.) but each transition can have another CF language within it.
  - Avantages:
    - Once the result is parsed, there's no need to do any additional interpretation
    - Avoids syntactic ambiguidies
    - Can tolerate noisy/non-grammatical input
  - Disadvantages:
    - Doesn't scale well
    - Doesn't do syntax generalization
- Augmented Transition Networks (ATNs)
  - Adds memory ("registers") to the network and allows links to manipulate the data structures.
  - Turing equivalent/universal programming language
  - In an ATN, a link can:
    - Check if an element is a member of a lexical category
    - Invoke a different sub-network
    - Set or read registers

    - Backtracking is controlled by success/failure of tests and the network structure
- Case Grammars
    - There are 8 different roles that a verb can be: goal, time, agent, instrument, etc.
    - Looks for the words that play specific roles in a sentence, and builds meaning from there
- Semantic Primitives
    - *There's going to be one CD question on the exam.*
    - Distill words to their basic meanings
    - Ideally, you could read into semantic primitives in one language then read out in another language
    - Like Schank's Conceptual Dependency
        - Reduced 500 verbs to 11 general actions:
            - ATRANS: transfer of abstract relationship
            - PTRANS: transfer of physical location of an object
            - PROPEL: application of physical force to an object
            - MOVE: movement of a body part by its owner
            - GRASP: grasping of an object by an actor
            - INGEST: ingestion of an object by an animal
            - EXPEL: expulsion of something from the body of an animal
            - MTRANS: transfer of mental information
            - MBUILD: building new information from existing information
            - SPEAK: production of sounds
            - ATTEND: Focusing of sensing organ towards a stimulus
        - CD could represent actions nearly canonically, and inferences could deal with primitives

        - Schank built SCRIPTS to represent stories, which were sequences of CD trees with variables. Programs could fill in the blanks to produce a complete story.
        - CD was very effective in the 1970s, and Schank et al built lots of impressive tools with them
        - CD was good for summarization
- Diversion into Siri (original version, pre-Apple acquisition, looking at a pitch deck)
    - Not built with neural nets, built with semantic parsing
    - Valuable because of ecosystem and integration of lots of different services
- Important distinction: shallow ML/deep learning NLP vs. true deep NLP
    - ML is growing increasingly popular, and is shockingly effective