3.4 Semantic Nets

- Developed to have (partial) graphical representation of predicate logic with special interpreted symbols
- First used to represent sentences in natural language
- Later abstracted to represent just meanings (Conceptual Dependency)
- Many different approaches (for example FIPA agent model)
- Sometimes used for describing ontologies
- Often also coupled with a logic and the possibility to add formulas to description (KL-ONE)
- Models classes and instances

Basic data structures

- Nodes: describe concepts and instantiations (objects, actions)
- Arcs/links: describe dependencies, like isa, is-element, greater-than,… can be predefined and user-defined
- Modifiers: add constraints, roles, etc. to links

Semantics

- Provide fixed interpretations for as many links as possible
- Provide fixed interpretations for modifiers
- User defined links require way to define their semantics (e.g. axioms in a logic with already defined semantics, or other descriptions)

Example: Conceptual Dependency (I)

- Actors: name or class name
- Actions (selection; including semantics):
  - ATRANS: Transfer of abstract relationship (give)
  - PTRANS: Transfer of physical location of object (go)
  - MOVE: Movement of body part by owner (kick)
  - INGEST: Ingesting of object by actor (eat)
  - MTRANS: Transfer of mental information (tell)
  - MBUILD: Building new information out of old (decide)

Example: Conceptual Dependency (II)

- Links:
  - \( \rightsquigarrow \) relation between actor and action
  - \( \leftarrow \rightarrow \) indicates dependency and direction of it
- Modifiers (selection; including semantics):
  - p: past tense
  - f: future
  - nil: present
  - for dependencies:
    - oc: object of an action
    - R: recipient of object
    - + user-defined modifiers

How to get knowledge into the representation structure

- Knowledge engineer should use as many predefined concepts, links and modifiers as possible in his/her graphs
- Knowledge engineer has to provide semantics (procedural, descriptive) for all user defined concepts, links and modifiers
Discussion

+ Semantic nets express structure in a way also understandable by humans
+ Easy to combine with other representation concepts
+ Easily extendable
  - Problem with how to express semantics for user-defined elements
  - Some extensions are not decidable
  - Often the predefined elements are not what we want for an application

And what about processing data?

- Answering questions:
  match question graph (with holes/variables) against graphs in knowledge base and return substitutions
  ⇓ search (for best match)
- Adding to existing knowledge-base (classification):
  match new knowledge against old and add new graph parts (while checking fulfillment of constraints)
  ⇓ search (for best fit)
- Other tasks:
  use provided procedures (based on semantics)
  for example: inference rules for CD actions

Examples

- Build a conceptual dependency representation for the following sentences:
  - John eats a steak
  - John ate pizza yesterday
- Build the graph for the following question and match it against the knowledge base from above:
  - Who had pizza yesterday?