

Knowledge Representation: Extended Example

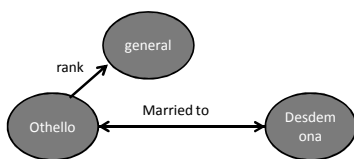
Trevor Bench-Capon
Room 215, Ashton Building

<http://www.csc.liv.ac.uk/~tbc/COMP210>

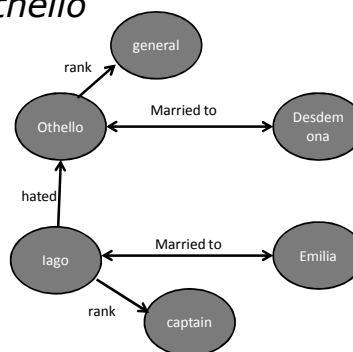
The Story of Othello

- Othello was a general who was married to Desdemona
- Iago was a captain who was married to Emilia. He hated Othello
- Iago told Othello lies about Desdemona.
- Othello killed Desdemona with a pillow. He felt remorse and killed himself with a dagger.

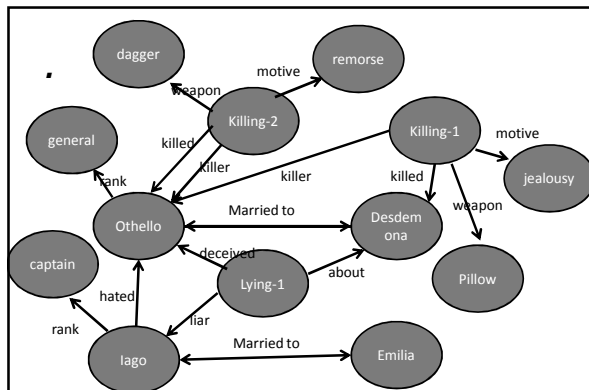
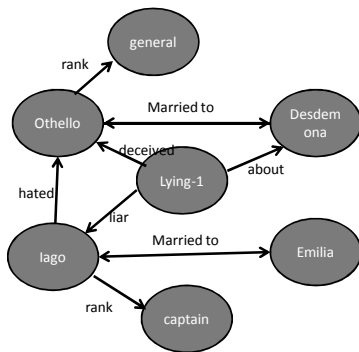
Othello was a general who was married to Desdemona



Iago was a captain who was married to Emilia. He hated Othello



Iago told Othello lies about Desdemona.



Othello killed Desdemona with a pillow. He felt remorse and killed himself with a dagger.

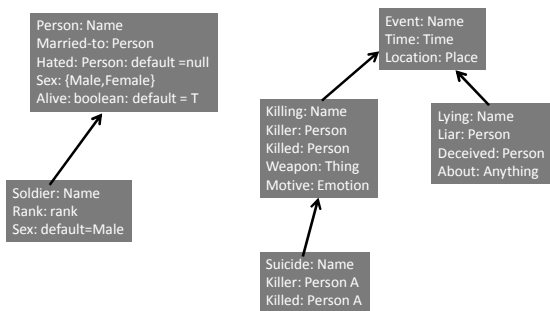
Entity Attribute Value Triples

- Othello rank general
- Othello married-to Desdemona
- Desdemona married-to Othello
- Iago rank captain
- Iago married-to Emilia
- Iago hated Othello
- Lying-1 liar Iago
- Lying-1 deceived Othello
- Lying-1 about Desdemona
- Killing-1 killer Othello
- Killing-1 killed Desdemona
- Killing-1 weapon pillow
- Killing-1 motive jealousy
- Killing-2 killer Othello
- Killing-2 killed Othello
- Killing-2 weapon dagger
- Killing-2 motive remorse
- Two kinds of entity: people and events

Organising into Frames

- People:
 - All have married-to
 - Two have rank
 - One has hated
- Events
 - Killings and Lyings have no attributes in common

Frame system



Script

Killing:
 Killer: Person1: question; "Who did the killing?"
 Killed: Person2: question; "Who was killed?"
 Weapon: Thing: question; "What weapon was used?"
 Motive: Emotion: question; "Why did Person1 kill Person2?"

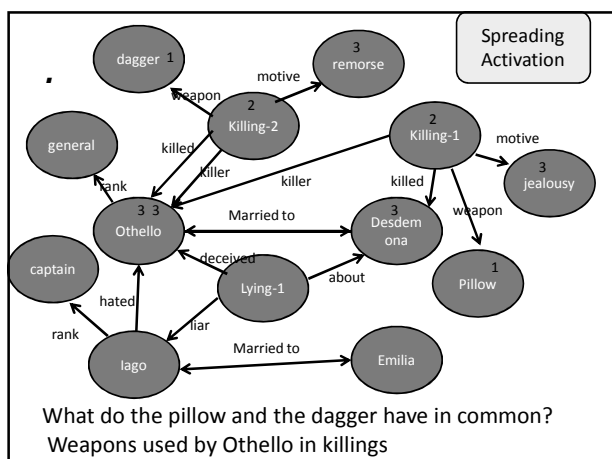
Given a frame with only some slots filled, we can ask questions to get more information

Prolog – organised by relations

marriedTo(Husband,Wife).
 marriedTo(X,Y):-marriedTo(Y,X).
 rank(Soldier,Rank).
 male(Person).
 alive(Person).
 killing(Killer,Killed,Weapon,Motive).
 motiveForKilling(Person,Motive):-
 killing(Person,_,_,Motive).
 And so on.

Manipulating the Knowledge

- So far we represented the knowledge in a variety of ways
- We also need to manipulate the knowledge
- This can be done in a variety of ways



Using Rules

- IF (?X is-a killing) AND (?X killing ?Y) THEN REMOVE (?Y alive T) AND ADD (?Y alive F).
- IF create(killing, ?X, ?Y) THEN execute(?X.weapon) AND execute(?X.motive) AND put(?Y.alive,F).
- Or we can use clauses for Prolog
 - alive(X,false):-killing(_X,_,_).

Developments

- Frame system + procedures for retrieving and manipulating knowledge = Object System
- Object Orientated Programming developed from Frame systems and has become a standard paradigm
- In Object Orientated Programming we use the procedural reading: in AI object are intended to model or simulate the domain
- OO Programming is a good example of how AI contributes to mainstream computing

Agents

- Agents can be seen as a development from OO programming:
 - Agents don't wait for messages: the proactively poll the environment to find new information
 - Agents decide whether to respond to messages
 - The elements of proactivity and autonomy make them (still) part of AI

Summary

- We have looked at an example showing how knowledge can be represented in a variety of ways
- Structured objects developed into OO programming, now a conventional technique
- Next:
 - Expert Systems as an Example of Rules
 - Ontologies as an extension of Structured Objects

Class Test

- Tomorrow Friday 19th March, Lecture Theatre D
- Counts 10% towards overall mark
- Questions on
 - Querying a Prolog program
 - Extending a Prolog Program
 - Lists and Recursion
- 50 Minutes will be allowed: it will start at 10.05 sharp
- Open Book: you may consult any written materials you wish (no computers).