Vorlesung Künstliche Intelligenz Wintersemester 2006/07

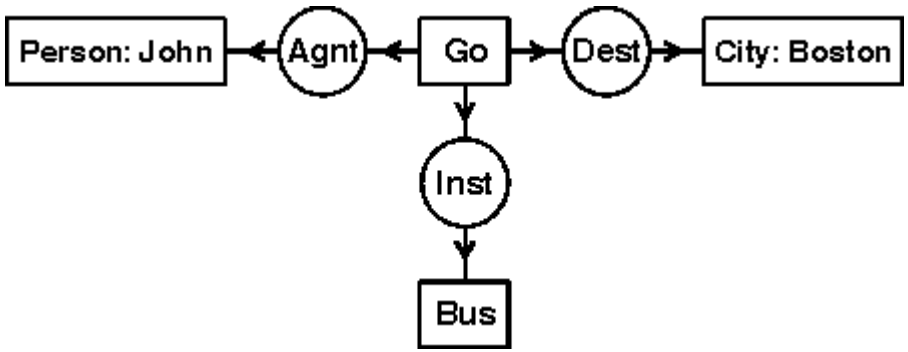
Teil III:
Wissensrepräsentation und Inferenz

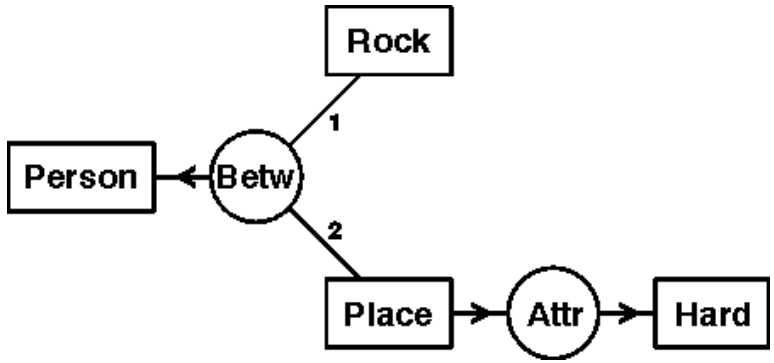
Kap.9: Begriffliche Graphen

Mit Material von John Sowa <http://www.jfsowa.com/cg/index.htm>
und Henrik Schärfe <http://www.huminf.aau.dk/cg/> .

Begriffliche Graphen

wurden 1984 von John Sowa eingeführt:





Begriffliche Graphen / Conceptual Graphs

[John]←(agnt)←[love]→(ptnt)→[Mary]

- Ein **begrifflicher Graph** ist ein beschrifteter bipartiter Graph:
  - *Begriffsknoten* repräsentieren Begriffe, Merkmale, oder Ereignisse.
  - *Relationsknoten* beschreiben Beziehungen zwischen den Begriffsknoten.
- **Alternative Definition:** Ein begrifflicher Graph ist ein ecken- und kantenbeschrifteter Multi-Hypergraph:
  - Die Knoten repräsentieren die Begriffe. Die Kanten beschreiben die Beziehungen.
  - Beziehungen wie “zwischen” können mehr als zwei Begriffe verbinden, deswegen “Hyper”.
  - Knoten können mit mehreren Kanten verbunden sein, deswegen “Multi”.

Definition of a concept

A concept is always made up of two entities:

- its *concept type*
- its *referent*

Concepts with both a type and a referent

The concept is drawn like this:

[*Type*: *Referent*]

where the type and the referent are separated by a colon.
For example:

[Person: John]

"There exists a person whose name is John„.

Here, "Person" is the concept type, and "John" is the referent.

Concepts with only a type

If the referent is blank, the concept is drawn like this:

[*Type*]

For example:

[Bus]
"There exists a bus"

The type can never be blank.

Concept Types

The *concept type* of a concept says what *kind* of concept we are dealing with. For example, these could all be types in our ontology:

Entity > Bus, Person, Tree, Location, Act, Animal.
Person > Student, Employee.
Employee > Professor.
Tree > SycamoreTree.
Location > City.
Act > Go, Leave, Eat, Catch.
Animal > Cat, Dog, Bird, Mouse.

What is a type?

A type is a *label* or *name* we give to a *group* of entities with similar traits. If we can categorize a number of individuals (e.g., "John", "Alfred", "Mary") in the same group (e.g., "Person"), then we can call the name of the group, together with the definition of the group, a "type".

Subtypes and supertypes

A type can be a subtype of another type. For example, in the above ontology, "Cat" is a subtype of "Animal", while "Student" is a subtype of "Person".

Referents

An *individual* from any of the groups that these types define can be the *referent* of a concept.

For example:

[Person: John]

"There exists a person whose name is John".

"Person" is the type, while "John" is the referent. In other words, "John" is the particular *individual* (or referent) which we are referring to, and he is an *instance* of the *type* "Person".

Blank referents

The referent may be left blank, as in the following conceptual graph:

[Bus]->(Dest)->[City: Aalborg]

Here, the concept "[Bus]" has no referent. The concept simply means "A bus" or "There is a bus".

Anytime the referent is left blank, it means "There is an X" or simply "An X", where "X" is the concept type.

The whole conceptual graph means:

"There is a bus which has a destination (Dest) which is a City, the referent of which is Aalborg."

Relations

A *relation type* is simply a *name* which we give to the relation.

It tells us what kind of relation we are dealing with. The type also *determines* the valence of the relation (ie, the number of arcs that belong to it) and its signature (the types of the concepts that are attached to those arcs).

For example, all of these are relation types:

- On (on)
- In(in)
- Dest(destination)
- Agnt(agent)
- Thme(theme)
- Ptnt(patient)
- Rcpt(recipient)

A very important relation is "Agnt" or "agent". It relates an act (such as "Sing") and an animate being (such as "Bird") which performs the act.

Example

[Sing]->(Agnt)->[Bird]->(In)->[SycamoreTree]

This conceptual graph says:

"There is a bird which is the agent of Sing. This same bird is in a sycamore tree".

Or, put into better English:

"A bird is singing in a sycamore tree".

- This graph has two relations, "Agnt" (agent) and "In" (in).
- The valence of Agnt is 2, and it relates an act with an animate being.
- The valence of In is 2, and it relates two physical entities in a spatial relationship.

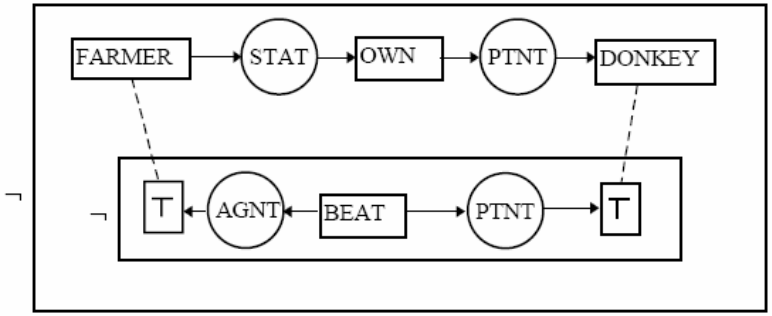
[Sing]->(Agnt)->[Bird]->(In)->[SycamoreTree]

$\exists$ x,y,z: Sing(x) $\wedge$ Bird(y) $\wedge$ SycamoreTree(z) $\wedge$ Agnt(x,y) $\wedge$ In(y,z)

- Die Semantik der Begrifflichen Graphen (soweit wir sie bisher kennen!) lässt sich durch direkte Übersetzung in die Prädikatenlogik bewerkstelligen.
- Wir haben bisher keine Negation verwendet, und kein Oder.
- Logische Inferenz wird auf dieser Ebene durch Graph Matching implementiert.

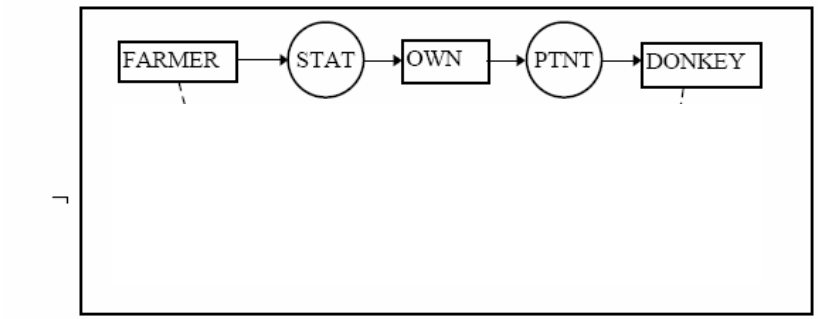
Komplexere Konstrukte: Negation

- Wie ist dieser Graph zu lesen?



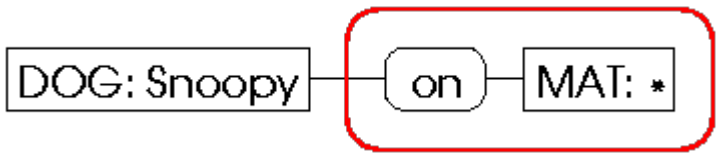
Hinweis: $\neg (A \wedge \neg B) \Leftrightarrow \neg A \vee B \Leftrightarrow A \rightarrow B$

Komplexere Konstrukte: Negation

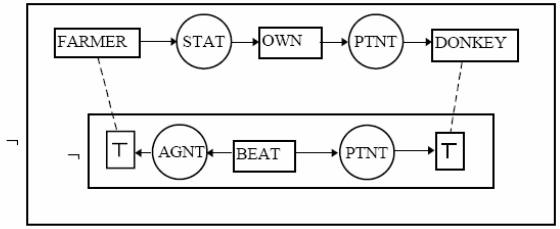


- Wie ist dieser Graph zu lesen?
- Was ist die intendierte Aussage?
- Stimmen beide überein?

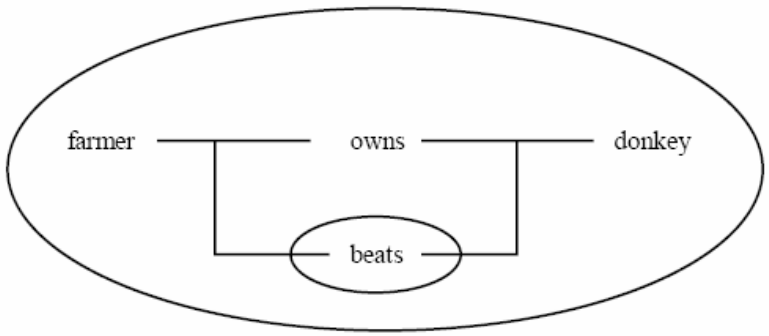
- Lösung durch F. Dau: partielle Cuts



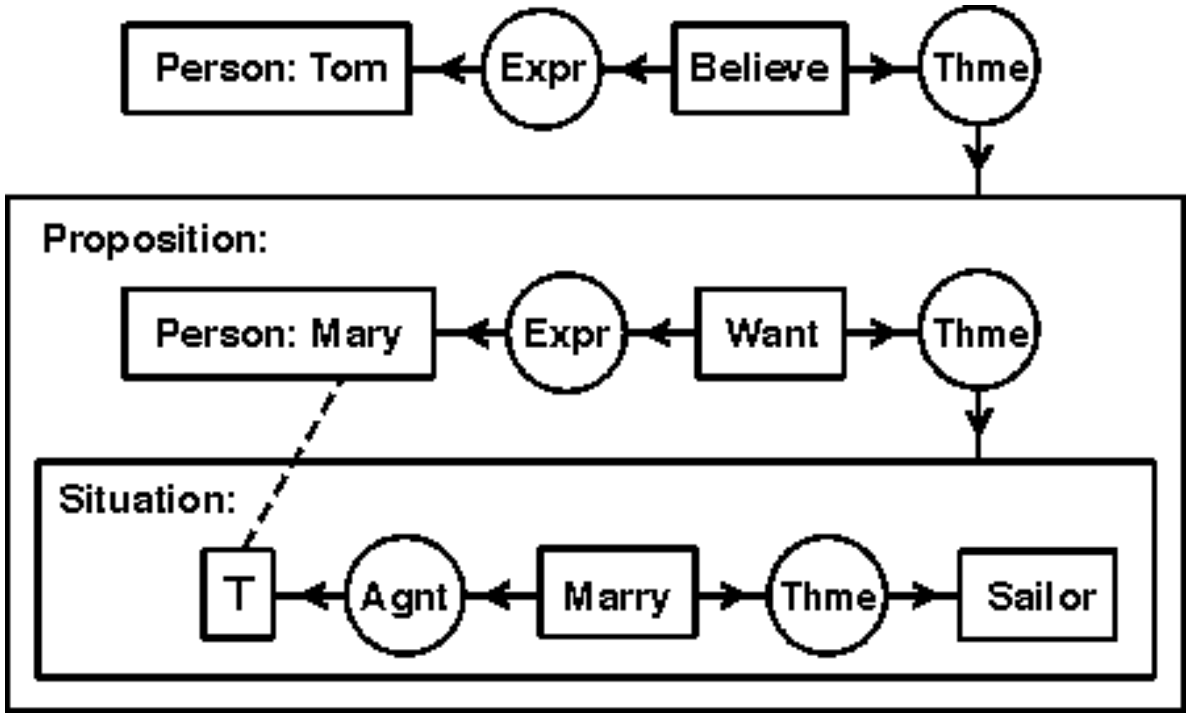
Exkurs: Historische Grundlage



- Charles S. Peirce: Existential Graphs (Ende 19. Jdt.)
- graphische Operatoren (z.B. Negation als „Cut“)
- Vorläufer der Prädikatenlogik
- umfasst Prädikatenlogik, Modallogik, etc.



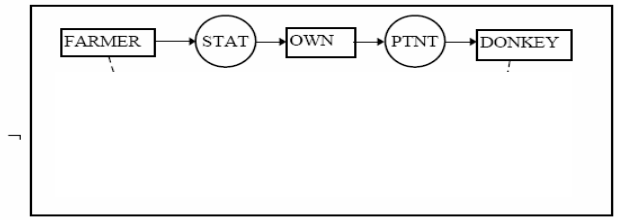
Nested Graphs / geschachtelte Graphen



- Charles S. Peirce: Existential Graphs (Ende 19. Jdt.)
- graphische Operatoren (z.B. Negation als „Cut“)
- Vorläufer der Prädikatenlogik
- umfasst Prädikatenlogik, Modallogik, etc.

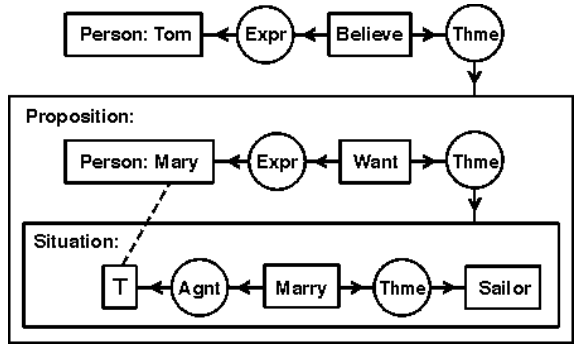
Vorteile Begrifflicher Graphen

- Intuitive Visualisierung



- Nähe zur natürlichen Sprache (z.B. was den Gültigkeitsbereich von Variablen in Wenn-Dann-Sätzen betrifft).

- Für einfache Graphen (dh ohne Negation und Schachtelung):
  - Nähe zur Prädikatenlogik
  - Dadurch klar definierte Semantik
  - Inferenz durch Graph Matching



- Methoden zur Definition komplexerer Begriffe/Relationen (λ-Kalkül)

Nachteile Begrifflicher Graphen / Semantik Teil 2

Es gibt keine formale Definition

- … der Semantik der Negation (abgesehen von der von Dau, die wenig bekannt ist).
- … für geschachtelte Graphen.

Es gibt also keine Garantie, dass derselbe Graph in unterschiedlichen Anwendungen gleich interpretiert wird!