



مجموعة جامعة
الجميع الذكية
للتعليم القابضة



Master of Science in Artificial intelligent

Lecture 8

AI511

Foundations of Artificial Intelligence

Prepared by
Prof. Lamiaa Fattouh Ibrahim

www.midocean.ae

OUTLINES

- Propositional Logic (Propositional Calculus)
 - The Language of Propositions
 - Connectives
 - Truth Values
 - Truth Tables
 - Applications
 - Translating English Sentences

OUTLINES

- **Predicate Logic (First-Order Logic (FOL), Predicate Calculus)**

The Language of Quantifiers

Translation from Predicate Logic to English

Translation from English to Predicate Logic

OUTLINES

Conceptual Graphs

- Definition of Conceptual Graphs
- Basic building blocks
- Concept node representation
- Exercise

Knowledge Representation

There are many methods used to knowledge representation for example:

1. Production Rule
2. Frames Representation
3. Propositional Calculus (Logic)
4. Predicate Calculus (Logic)
5. Conceptual Graph
6. Semantic Network
7. Script Representation

Propositional Calculus

- A proposition is a statement or an argument that can be True or False.
- A **proposition** is a declarative sentence (that is, a sentence that declares a fact) that is either true or false, but not both.
- Propositional logic is a representation language for AI.
- Using **words**, **phrases**, and **sentences**, we can **represent** and **reason** about **properties** and **relationships** in the world.

Propositions

- A *proposition* is a declarative sentence that is either **true** or **false**.
- Examples of propositions:

a) The Moon is made of green cheese.	false
b) Trenton is the capital of New Jersey.	true
c) Toronto is the capital of Canada.	false
d) $1 + 0 = 1$	true
e) $0 + 0 = 2$	false
- Examples that are not propositions.

a) Sit down!	Sentences a and b are not propositions because they are not declarative sentences. Sentences c and d are not propositions because they are neither true nor false. X is variable. Note that each of sentences c and d can be turned into a proposition if we assign values to the variables.
b) What time is it?	
c) $x + 1 = 2$	
d) $x + y = z$	

PROPOSITIONAL CALCULUS SYMBOLS

- Propositional symbols denote propositions, or statements about the world that may be either true or false, such as “the car is red” or “the street is wet.”
- The symbols of propositional calculus are:
- Propositional symbols: $P, Q, R, S, \dots$
- Truth symbols: $\text{true}, \text{false}$,
- Connectives: $\wedge, \vee, \neg, \rightarrow, \equiv$

PROPOSITIONAL CALCULUS SENTENCES

- Every **propositional symbol** and **truth symbol** is a sentence.
- For example: true, P, Q, and R are sentences.

Negation

- The **negation** of a sentence is a sentence.
- For example: $\neg P$ and $\neg \text{false}$ are sentences.

Conjunction

- The **conjunction**, or and, of two sentences is a sentence.
- For example: $P \wedge Q$ is a sentence.
- P and Q are called **Conjuncts**.

Disjunction

- The **disjunction**, or or, of two sentences is a sentence.
- For example: $P \vee Q$ is a sentence.
- P and Q are called **Disjuncts**.

Implication

- The **implication** of one sentence from another is a sentence.
- For example: $P \rightarrow Q$ is a sentence.
- P is called **premise / antecedent**.
- Q is called **conclusion / consequent**.

Equivalence

- The **equivalence** of two sentences is a sentence.
- For example: $P \vee Q \equiv R$ is a sentence.

Grouping of sentences

- In propositional calculus sentences, the symbols $()$ and $[]$ are used to group symbols into sub-expressions and so to control their order of evaluation and meaning.
- $(P \vee Q) \equiv R$ is different from $P \vee (Q \equiv R)$

Well Formed Formula

- Legal sentences are also called well-formed formulas or **WFFs**.
- An expression is a sentence, or well-formed formula, of the propositional calculus **if and only if** it can be formed of legal symbols through some sequence of these rules.

Example

- $((P \wedge Q) \rightarrow R) \equiv \neg P \vee \neg Q \vee R$
- is a well-formed sentence in the propositional calculus because:
- P , Q , and R are propositions and thus sentences.
- $P \wedge Q$, the conjunction of two sentences, is a sentence.
- $(P \wedge Q) \rightarrow R$, the implication of a sentence for another, is a sentence.
- $\neg P$ and $\neg Q$, the negations of sentences, are sentences.
- $\neg P \vee \neg Q$, the disjunction of two sentences, is a sentence.
- $\neg P \vee \neg Q \vee R$, the disjunction of two sentences, is a sentence.
- $((P \wedge Q) \rightarrow R) \equiv \neg P \vee \neg Q \vee R$, the equivalence of two sentences, is a sentence.
- We constructed the original expression through a series of applications of **legal** rules and is therefore “**well formed**”.

Semantics

- The semantic of a statement is its **meaning**
- A proposition must be either true or false, given some state of the world.
- The truth value assignment to propositional sentences is called an **interpretation**, which is an assertion about their truth in some possible world.
- Formally, an interpretation is a mapping from the propositional symbols into the set $\{T, F\}$.

Example

- if P denotes the proposition “it is raining” and
- Q denotes “I am at work,” then
- the set of propositions $\{P, Q\}$ has four different functional mappings into the truth values $\{T, F\}$.
- These mappings correspond to four different interpretations.

PROPOSITIONAL CALCULUS SEMANTICS

- An interpretation of a set of propositions is the assignment of a truth value, either T or F, to **each propositional symbol**.
- The truth symbol true is always assigned T, and
- the symbol false is always assigned F.

Negation

The truth assignment of negation, $\neg P$, where P is any propositional symbol, is F if the assignment to P is T, and T if the assignment to P is F.

How many rows are there in a truth table with n propositional variables?

Solution: 2^n

TABLE 1 The Truth Table for the Negation of a Proposition.

p	$\neg p$
T	F
F	T

Conjunction

- The truth assignment of conjunction, $\wedge$, is T only when both conjuncts have truth value T; otherwise it is F.

TABLE 2 The Truth Table for the Conjunction of Two Propositions.

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction

- The truth assignment of disjunction, $\vee$, is F only when both disjuncts have truth value F; otherwise it is T.

TABLE 3 The Truth Table for the Disjunction of Two Propositions.

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

Implication

The truth assignment of implication, $\rightarrow$, is F only when the premise or symbol before the implication is T and the truth value of the consequent or symbol after the implication is F; otherwise it is T.

TABLE 5 The Truth Table for the Conditional Statement $p \rightarrow q$.

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

Different Ways of Expressing $p \rightarrow q$

- if p , then q
- p implies q
- if p , q
- p only if q
- q unless $\neg p$
- q when p
- q if p
- q whenever p
- p is sufficient for q
- q follows from p
- q is necessary for p

a necessary condition for p is q

a sufficient condition for q is p

Equivalence

- The truth assignment of equivalence, $\equiv$, is T only when both expressions have the same truth assignment for all possible interpretations; otherwise it is F.

Converse, Contrapositive, and Inverse

- From $p \rightarrow q$ we can form new conditional statements .
 - $q \rightarrow p$ is the **converse** of $p \rightarrow q$
 - $\neg q \rightarrow \neg p$ is the **contrapositive** of $p \rightarrow q$
 - $\neg p \rightarrow \neg q$ is the **inverse** of $p \rightarrow q$

Example: Find the converse, inverse, and contrapositive of “It raining is a sufficient condition for my not going to town.”

Solution:

converse: If I do not go to town, then it is raining.

contrapositive: If I go to town, then it is not raining.

inverse: If it is not raining, then I will go to town.

Truth Tables

- A truth table lists all possible truth value assignments to the atomic propositions of an expression and gives the truth value of the expression for each assignment.
- A truth table enumerates all possible worlds of interpretation that may be given to an expression.

- Example:

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

Example Truth Table

- Construct a truth table for

$$p \vee q \rightarrow \neg r$$

p	q	r	$\neg r$	$p \vee q$	$p \vee q \rightarrow \neg r$
T	T	T	F	T	F
T	T	F	T	T	T
T	F	T	F	T	F
T	F	F	T	T	T
F	T	T	F	T	F
F	T	F	T	T	T
F	F	T	F	F	T
F	F	F	T	F	T

Equivalent Propositions

- Two propositions are *equivalent* if they always have the same truth value.
- Example:** Show using a truth table that the conditional is equivalent to the contrapositive.

Solution:

p	q	$\neg p$	$\neg q$	$p \rightarrow q$	$\neg q \rightarrow \neg p$
T	T	F	F	T	T
T	F	F	T	F	F
F	T	T	F	T	T
F	F	T	T	T	T

Using a Truth Table to Show Non-Equivalence

Example: Show using truth tables that neither the converse nor inverse of an implication are not equivalent to the implication.

Solution:

p	q	$\neg p$	$\neg q$	$p \rightarrow q$	$\neg p \rightarrow \neg q$ inverse	$q \rightarrow p$ converse
T	T	F	F	T	T	T
T	F	F	T	F	T	T
F	T	T	F	T	F	F
F	F	T	T	T	T	T

Exercise

- Prove that the statement $(P \rightarrow Q)$ is equivalent to the statement $(\neg P \vee Q)$ using truth tables.

Applications of Propositional Logic

Translating English Sentences

Once we have translated sentences from English into logical expressions we can analyze these logical expressions to determine their truth values, we can manipulate them, and we can use rules of inference to reason about them.

Translating English Sentences

- Steps to convert an English sentence to a statement in propositional logic
 - Identify atomic propositions and represent using propositional variables.
 - Determine appropriate logical connectives
- “If I go to Harry’s or to the country, I will not go shopping.”
 - p : I go to Harry’s
 - q : I go to the country.
 - r : I will go shopping.

If p or q then not r .

$$(p \vee q) \rightarrow \neg r$$

Example

Problem: Translate the following sentence into propositional logic:

“You can access the Internet from campus only if you are a computer science major or you are not a freshman.”

One Solution: Let a , c , and f represent respectively “You can access the internet from campus,” “You are a computer science major,” and “You are a freshman.”

$$a \rightarrow (c \vee \neg f)$$

Equivalences

- $\neg (\neg P) \equiv P$
- $(P \vee Q) \equiv (\neg P \rightarrow Q)$
- the contrapositive law: $(P \rightarrow Q) \equiv (\neg Q \rightarrow \neg P)$
- de Morgan's law: $\neg (P \vee Q) \equiv (\neg P \wedge \neg Q)$
- de Morgan's law: $\neg (P \wedge Q) \equiv (\neg P \vee \neg Q)$
- the commutative laws: $(P \wedge Q) \equiv (Q \wedge P)$
- the commutative laws: $(P \vee Q) \equiv (Q \vee P)$
- the associative law: $((P \wedge Q) \wedge R) \equiv (P \wedge (Q \wedge R))$
- the associative law: $((P \vee Q) \vee R) \equiv (P \vee (Q \vee R))$
- the distributive law: $P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$
- the distributive law: $P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$

- Identities such as these can be used to change propositional calculus expressions into a syntactically different but logically equivalent form.
- These identities may be used instead of truth tables to prove that two expressions are equivalent by finding a series of identities that transform one expression into the other.

Exercises

- Using identities, prove the following equivalences:
- $\neg (P \rightarrow Q) \equiv (P \wedge \neg Q)$
- $P \vee (P \wedge Q) \equiv P$

Predicate Calculus

..

Propositional vs Predicate Calculus

- There is no way to access the components of an individual assertion using propositional calculus, while we can do this in predicate calculus.
- Through inference rules we can manipulate predicate calculus expressions, accessing their individual components and inferring new sentences.
- Predicate calculus also allows expressions to contain variables. Variables let us create general assertions about classes of entities.

PREDICATE CALCULUS ALPHABET

- The alphabet that makes up the symbols of the predicate calculus consists of:
 1. The set of letters, both upper- and lowercase, of the English alphabet.
 2. The set of digits, 0, 1, ..., 9.
 3. The underscore, _.
- **Symbols begin with a letter and are followed by any sequence of these legal characters.**

Predicate Calculus SYMBOLS

- Predicate calculus symbols include:
 1. **Truth symbols** true and false (these are reserved symbols).
 2. **Constant symbols** are symbol expressions having the first character lowercase.
 3. **Variable symbols** are symbol expressions beginning with an uppercase character.
 4. **Function symbols** are symbol expressions having the first character lowercase.
 - Functions have an attached **arity** indicating the number of elements of the domain mapped onto each element of the range.
 - A function expression consists of a function constant of arity n, followed by n terms, enclosed in parentheses and separated by commas.
 - The act of replacing a function with its value is called **evaluation**.

TERMS

- A predicate calculus **term** is either a **constant**, **variable**, or **function expression**.
- Term may be used to denote objects and properties in a problem domain.

PREDICATES

- Predicate symbols are symbols beginning with a lowercase letter.
- Predicates have an associated positive integer referred to as the **arity** or “argument number” for the predicate.
- **Predicates with the same name but different arities are considered distinct.**

ATOMIC SENTENCES

- An atomic sentence is a **predicate** constant of arity n , followed by n terms, enclosed in parentheses and separated by commas.
- The truth values, **true** and **false**, are also atomic sentences.
- Atomic sentences are also called **atomic expressions**, **atoms**, or **propositions**.

Examples of atomic sentence are:

likes(george,kate)

likes(george,susie)

likes(george,sarah,tuesday)

friends(bill,george)

helps(bill,george)

likes(X,george)

likes(X,X)

friends(bill,richard)

friends(father_of(david),father_of(andrew))

helps(richard,bill)

In the predicates above, bill, george, kate, etc., are constant symbols and represent objects in the problem domain. The arguments to a predicate are terms and may also include variables or function expressions. For example,

friends(father_of(david),father_of(andrew))

PREDICATE CALCULUS SENTENCES

- Every atomic sentence is a sentence.
- 1. If s is a sentence, then so is its **negation**, $\neg s$.
- 2. If s_1 and s_2 are sentences, then so is their **conjunction**, $s_1 \wedge s_2$.
- 3. If s_1 and s_2 are sentences, then so is their **disjunction**, $s_1 \vee s_2$.
- 4. If s_1 and s_2 are sentences, then so is their **implication**, $s_1 \rightarrow s_2$.
- 5. If s_1 and s_2 are sentences, then so is their **equivalence**, $s_1 \equiv s_2$.
- 6. If X is a variable and s a sentence, then $\forall X s$ is a sentence.
- 7. If X is a variable and s a sentence, then $\exists X s$ is a sentence.

Example

- mother(eve, abel)
- mother(eve, cain)
- father(adam, abel)
- father(adam, cain)
- $\forall X \forall Y \text{ father}(X, Y) \vee \text{ mother}(X, Y) \rightarrow \text{parent}(X, Y)$
- $\forall X \forall Y \forall Z \text{ parent}(X, Y) \wedge \text{parent}(X, Z) \rightarrow \text{sibling}(Y, Z)$

DEFINITION : INTERPRETATION

- Let the domain D be a nonempty set.
- An interpretation over D is an assignment of the entities of D to each of the constant, variable, predicate, and function symbols of a predicate calculus expression, such that:
 1. Each constant is assigned an element of D .
 2. Each variable is assigned to a nonempty subset of D ; these are the allowable substitutions for that variable.
 3. Each function f of arity m is defined on m arguments of D and defines a mapping from D^m into D .
 4. Each predicate p of arity n is defined on n arguments from D and defines a mapping from D^n into $\{T, F\}$.

TRUTH VALUE OF PREDICATE CALCULUS EXPRESSIONS

Assume an expression E and an interpretation I for E over a nonempty domain D .

The truth value for E is determined by:

1. The value of a constant is the element of D it is assigned to by I .
2. The value of a variable is the set of elements of D it is assigned to by I .
3. The value of a function expression is that element of D obtained by evaluating the function for the parameter values assigned by the interpretation.
4. The value of truth symbol “true” is T and “false” is F .
5. The value of an atomic sentence is either T or F , as determined by the interpretation I .

PREDICATE CALCULUS EXPRESSIONS

6. The value of the negation of a sentence is T if the value of the sentence is F and is F if the value of the sentence is T.
7. The value of the conjunction of two sentences is T if the value of both sentences is T and is F otherwise.
8. The truth value of expressions using $\vee$, $\rightarrow$, and $\equiv$ is determined from the value of their operands.
9. For a variable X and a sentence S containing X :
The value of $\forall X S$ is T if S is T **for all assignments to X** under I , and it is F otherwise.
10. The value of $\exists X S$ is T if there is **at least one** assignment to X in the interpretation under which S is T; otherwise it is F.

Examples

- If it doesn't rain on Monday, Tom will go to the mountains.
 $\neg \text{weather}(\text{rain}, \text{monday}) \rightarrow \text{go}(\text{tom}, \text{mountains})$
- Emma is a Doberman pinscher and a good dog.
 $\text{gooddog}(\text{emma}) \wedge \text{isa}(\text{emma}, \text{doberman})$
- All basketball players are tall. $\forall X (\text{basketball_player}(X) \rightarrow \text{tall}(X))$
- Some people like anchovies.
 $\exists X (\text{person}(X) \wedge \text{likes}(X, \text{anchovies}))$
- If wishes were horses, beggars would ride.
 $\text{equal}(\text{wishes}, \text{horses}) \rightarrow \text{ride}(\text{beggars})$
- Nobody likes taxes.
 $\neg \exists X \text{ likes}(X, \text{taxes})$

Definition of Conceptual Graphs

A conceptual graph consists of concept nodes and relation nodes

- The **concept nodes** represent entities, attributes, states, and events
 concept nodes represent either:
 - a- **Concrete concepts:** such as a cat, telephone, or restaurant, are characterized by our ability to form an image of them in our minds.
 - b- **abstract concepts:** include things such as love, beauty, and loyalty that do not correspond to image in our minds.
- The **relation nodes** show how the concepts are interconnected.

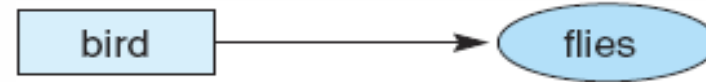
A Graph-Theoretic Definition

- Conceptual Graphs are **finite, connected, bipartite** graphs.
 - **Finite:** because any graph (in 'human brain' or 'computer storage') can only have a finite number of concepts and conceptual relations.
 - **Connected:** because two parts that are not connected would simply be called two conceptual graphs.
 - **Bipartite:** because there are two different kinds of nodes: **concepts** and **conceptual relations**, and every **arc** links a node of one kind to a node of another kind

- A relation of arity n is represented by a conceptual relation node having n arcs.
- Each conceptual graph represents a single proposition.
- A typical knowledge base will contain a number of such graphs.
- Graphs may be arbitrarily complex but must be finite.

Fig 7.14

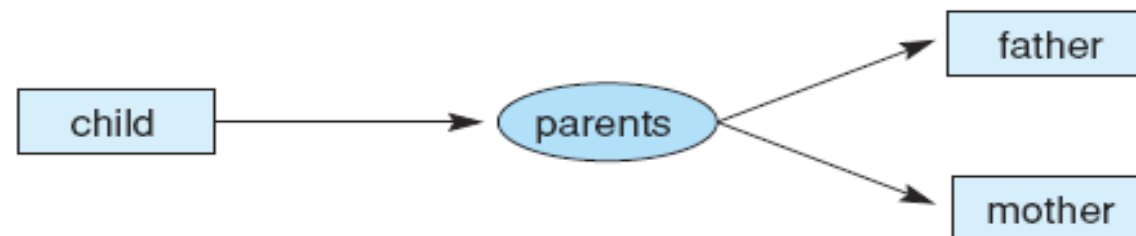
Conceptual relations of different arities.



Flies is a 1-ary relation.



Color is a 2-ary relation.

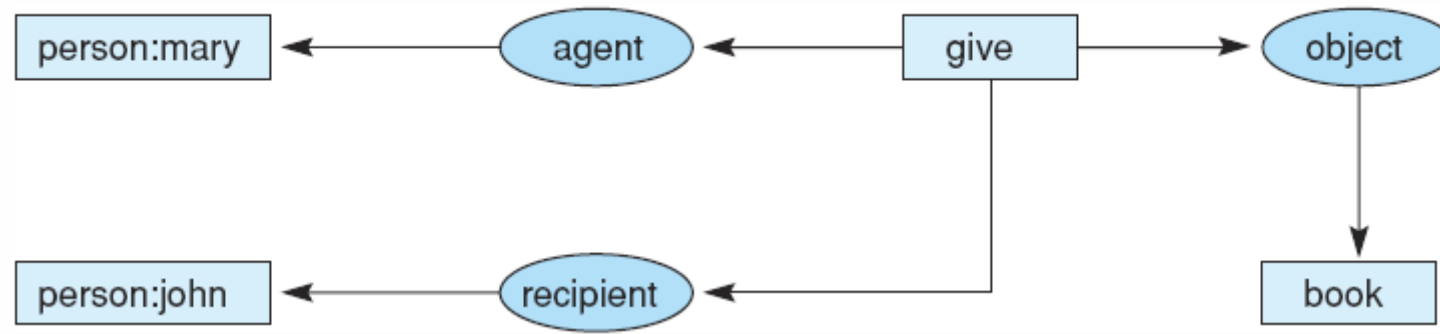


Parents is a 3-ary relation.

Concept Nodes

- In conceptual graphs CG, every concept is a unique individual of a particular type. Each box is labeled with a type label, which indicates the class or type of individual represented by that node.
- CG allows nodes to be labelled simultaneously with the name of the individual the node represents and its type. The two are separated by a colon (":")
- Boxes with the same type label represent concepts of the same type; however, these boxes may or may not represent the same individual concept.

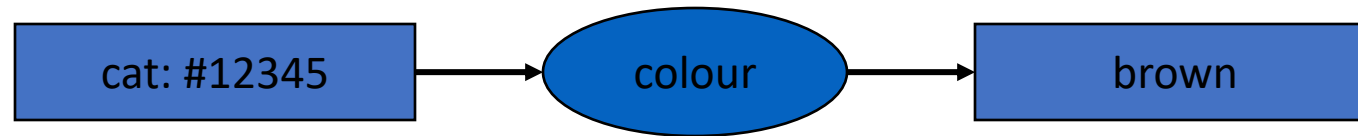
Fig 7.15 Graph of “Mary gave John the book.”



This graph uses conceptual relations to represent the cases of the verb “to give” and indicates the way in which conceptual graphs are used to model the semantics of natural language.

Concept Nodes: Unnamed Individuals

- Consider the example that we do not know the name of a cat that is brown:



- Each concept node in a CG may be used to represent specific but unnamed individuals.
- A unique token called a marker indicates each individual in the world of discourse.
- This marker is written as a number preceded by a #.

Concept Nodes: Multiple Names

- We subsequently found out that the cat is called by different names: "Sylvester", "Sugar Pie" and "Squidgy Bod". The name is enclosed in double quotes to indicate that it is a string.

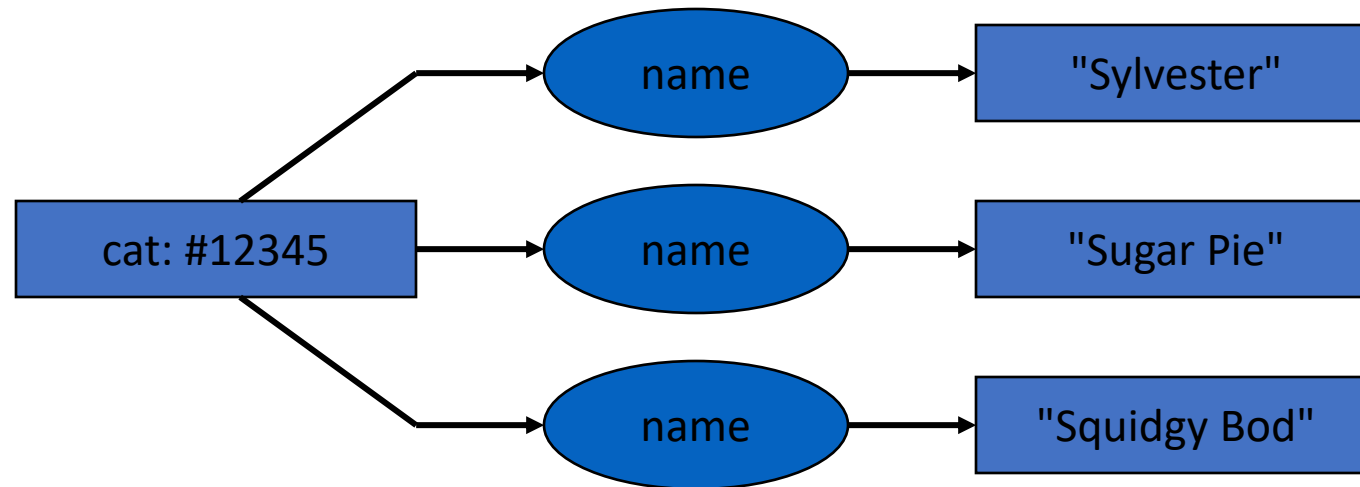
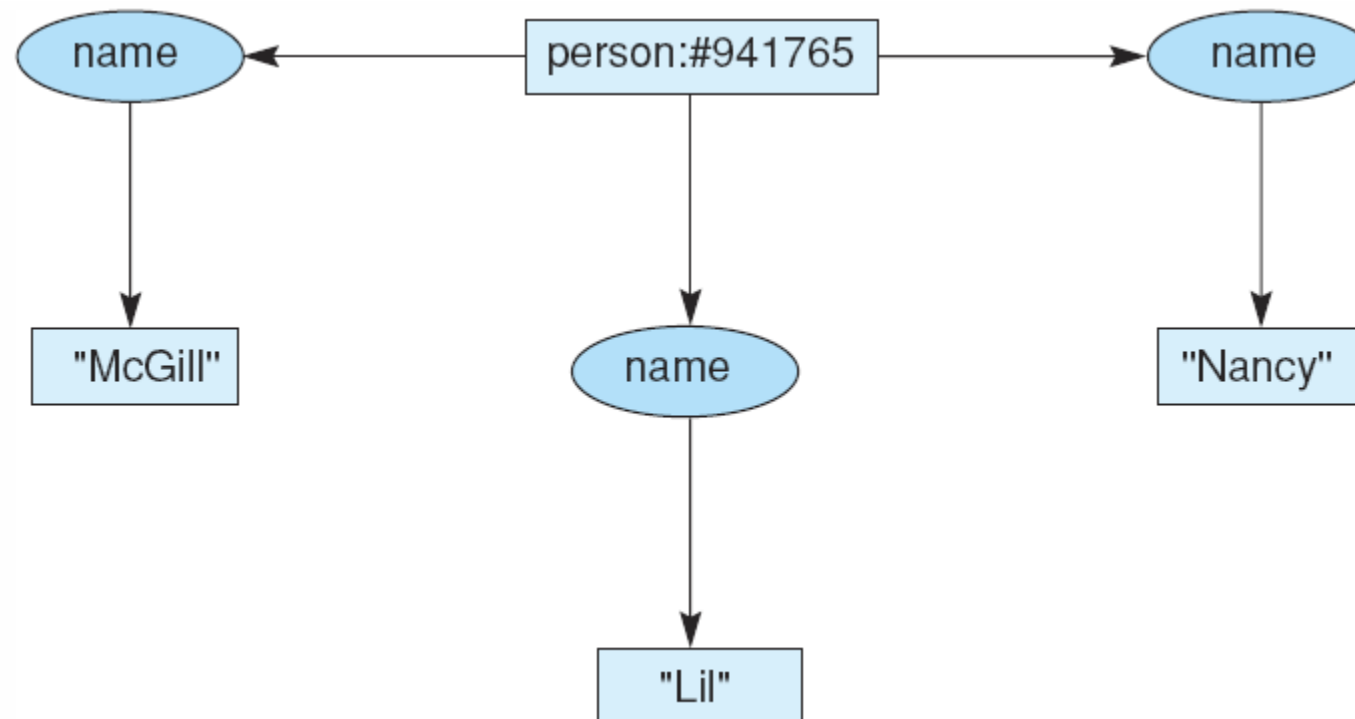
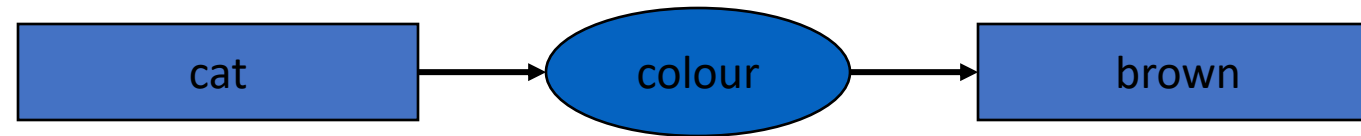


Fig 7.19 Conceptual graph of a person with three names.

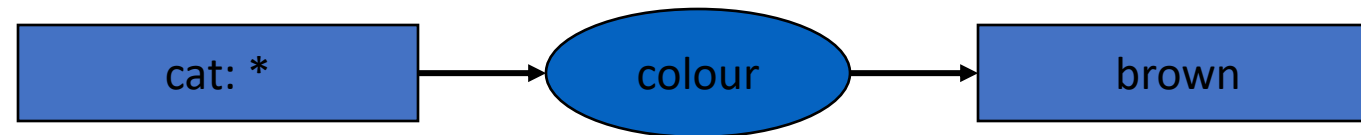


Concept Nodes: Unspecified Individuals

- General markers can also be used to refer to an unspecified individual. The CG:



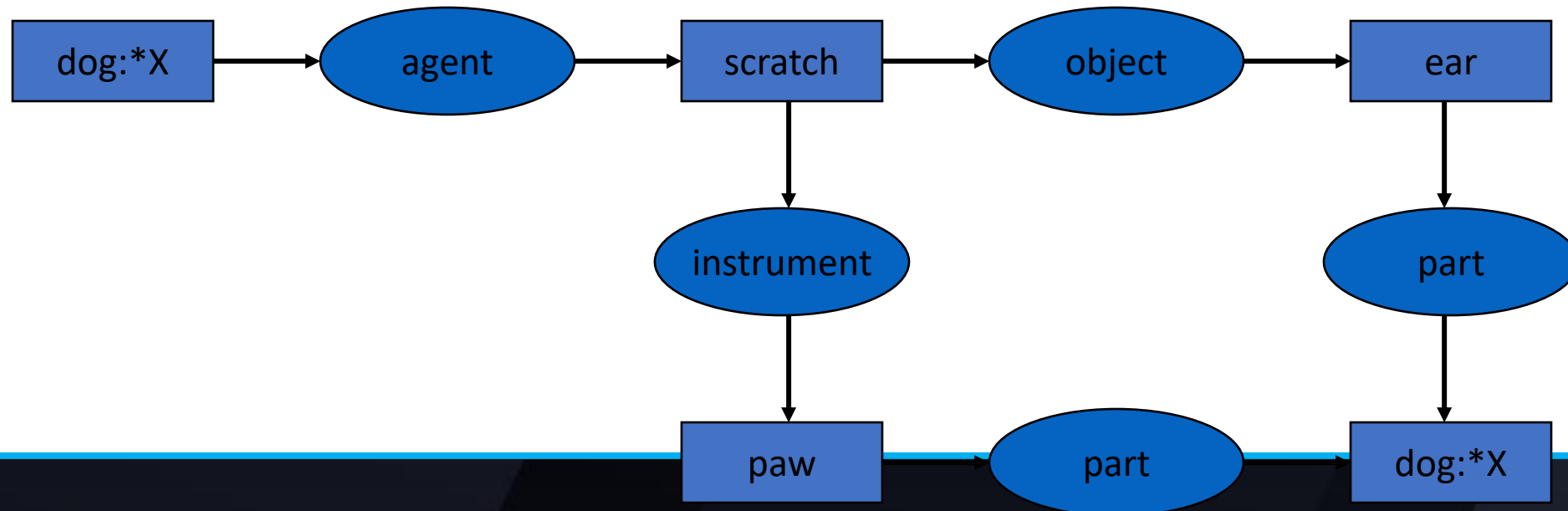
- Refers to an unspecified cat. Notationally, unspecified individuals are shown by the existence of an asterisk ("*")



- BUT... this is usually omitted (cat = cat:*)).

Concept Nodes: Named Variables

- Named variables can also be used to refer to an individual. These are represented by an asterisk followed by the variable name.
- This is useful to indicate nodes that are the same unspecified individual.



Exercises

- Please create the conceptual graph of the following sentence:
 - John is between a rock and a hard place

Solution 1

- "John is between a rock and a hard place"

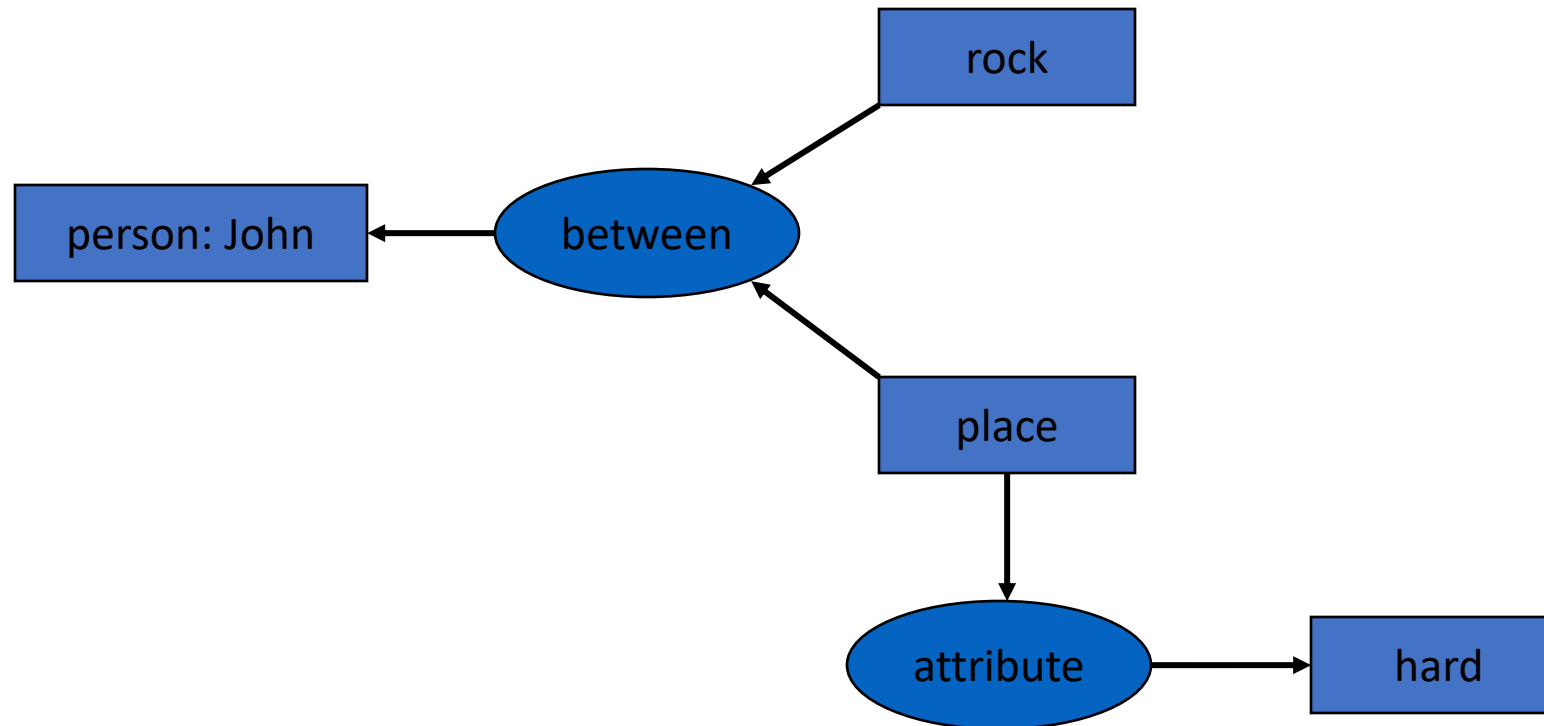


Fig 7.16 Conceptual graph indicating that the dog named Emma is brown.

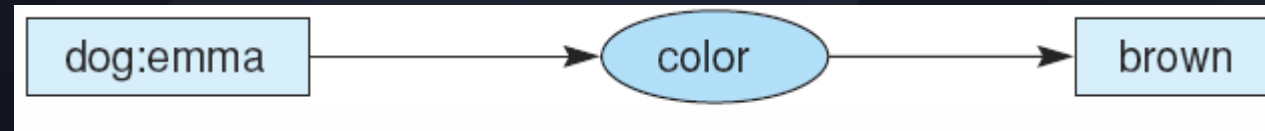
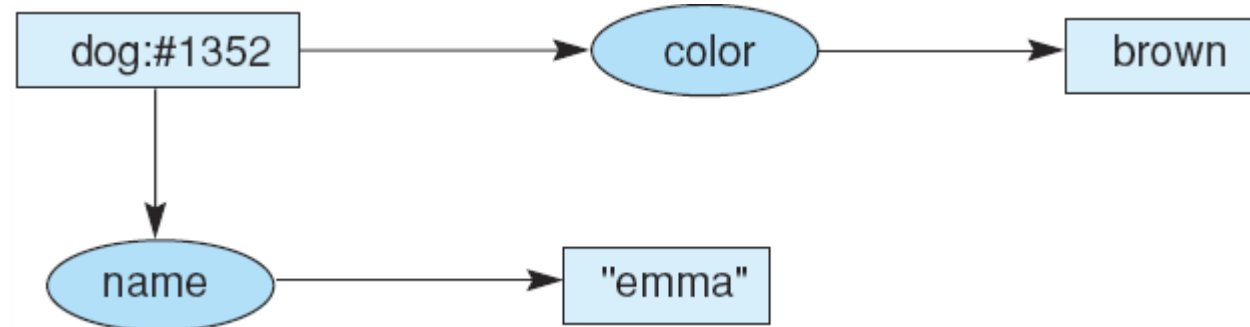


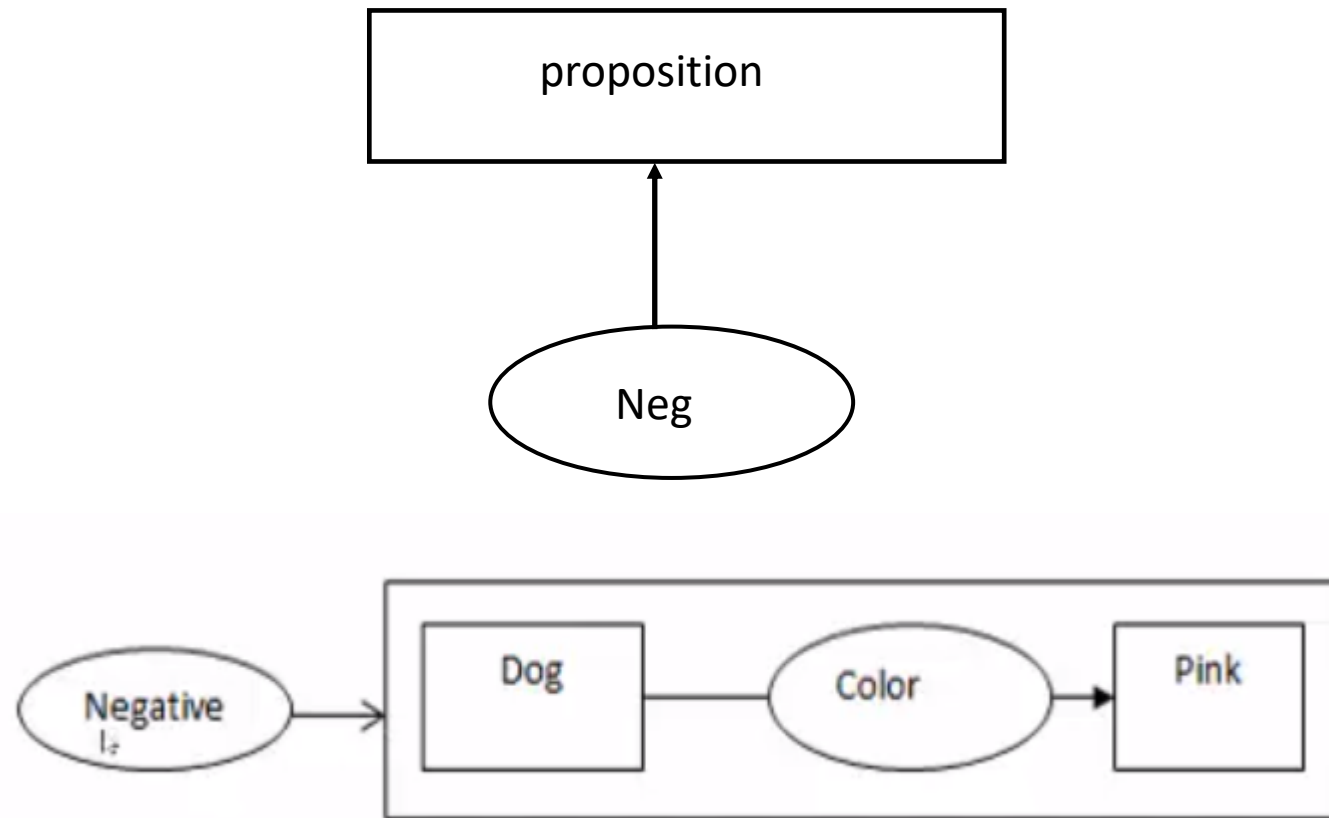
Fig 7.17 Conceptual graph indicating that a particular (but unnamed) dog is brown.



Fig 7.18 Conceptual graph indicating that a dog named Emma is brown.



We can simplify the graph and refer to the individual directly by name. under this convention, The graph of figure 7.18 is equivalent to the graph of figure 7.16



Summary

knowledge representation

- **Propositional Logic (Propositional Calculus)**
- **Predicate Logic (First-Order Logic (FOL), Predicate Calculus)**
- **Conceptual Graphs**



مجموعة جامعة
الجميع الذكية
للتعليم القابضة

MIDOCEAN
UNIVERSITY
BY EVERYONE'S SMART UNIVERSITY



Thank You

