Knowledge Representation and Reasoning

Introduction

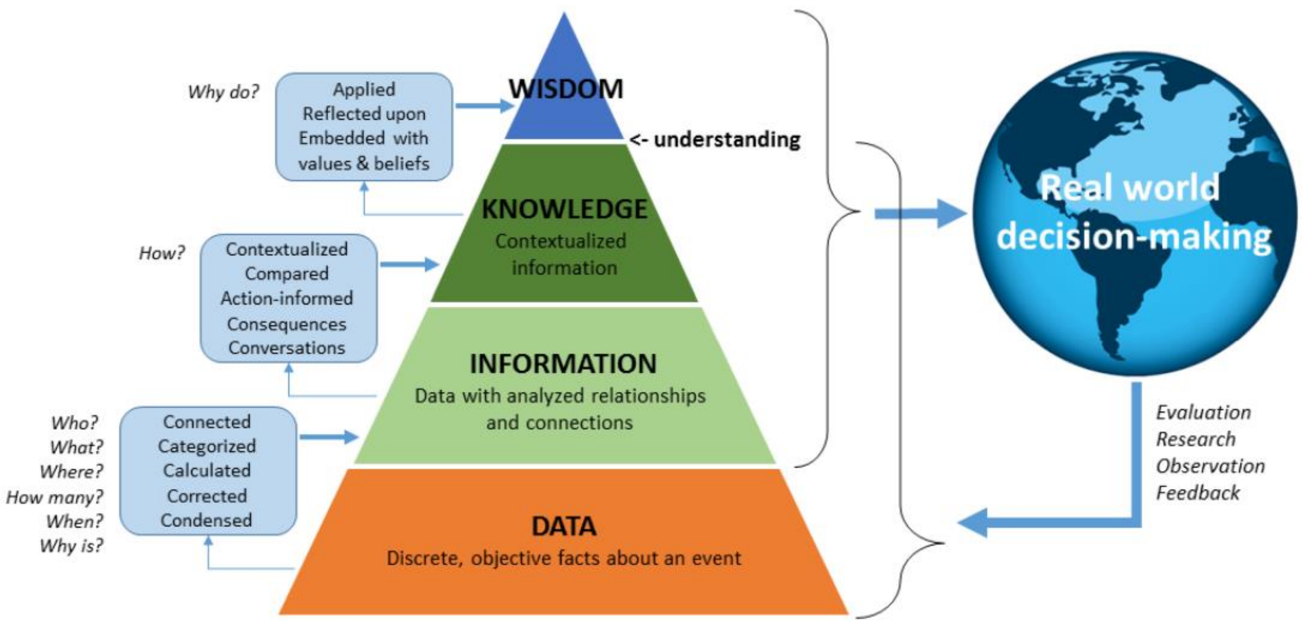
- Knowledge Representation (KR) is the field of artificial intelligence (AI) dedicated to representing information about the world in a form that a computer system can utilize to solve complex tasks such as diagnosing a medical condition
- Examples of knowledge representation models include logic, semantic nets, frames, scripts, and ontologies.
- Automated reasoning engines include Inference engines, theorem provers, classifiers etc

What is knowledge?

- Data: Data is raw facts about the world or events
- Information: Result of processing raw data
- Knowledge: Information gained through experience, reasoning, or acquaintance
- Examples: Driving your car, placement process at VIT-AP, etc.

Wisdom

- The ability to distinguish or judge what is true, right, or lasting
- Knowledge can exist without wisdom, but not the other way around
- Knowledge is knowing how to kill opponents in battlefield; wisdom is knowing what to do with an unarmed opponent



WISDOM
<- understanding
KNOWLEDGE
Contextualized information
INFORMATION
Data with analyzed relationships and connections
DATA
Discrete, objective facts about an event
Why do?
Applied
Reflected upon
Embedded with values & beliefs
How?
Contextualized
Compared
Action-informed
Consequences
Conversations
Who?
What?
Where?
How many?
When?
Why is?
Connected
Categorized
Calculated
Corrected
Condensed
Real world decision-making
Evaluation
Research
Observation
Feedback

Representation and Mapping

- Tommy is a dog
- Every dog has a tail

Tommy has a tail

Representation and Mapping

- Tommy is a dog

dog(Tommy)

- Every dog has a tail

$\forall x$: dog(x) $\rightarrow$ hastail(x)

hastail(Tommy)

Tommy has a tail

Propositional Logic

Propositional logic

- **Logical constants**: true, false
- **Propositional symbols**: P, Q, S, ... (**atomic sentences**)
- Wrapping **parentheses**: (…)
- Sentences are combined by **connectives**:

$\wedge$	...and	[conjunction]
$\vee$	...or	[disjunction]
$\Rightarrow$	...implies	[implication / conditional]
$\Leftrightarrow$	..is equivalent	[biconditional]
$\neg$	...not	[negation]

- **Literal**: atomic sentence or negated atomic sentence

Examples of PL sentences

- P means “It is hot.”
- Q means “It is humid.”
- R means “It is raining.”
- $(P \wedge Q) \rightarrow R$

  “If it is hot and humid, then it is raining”
- $Q \rightarrow P$

  “If it is humid, then it is hot”
- A better way:

  Hot = “It is hot”

  Humid = “It is humid”

  Raining = “It is raining”

Propositional logic (PL)

- A simple language useful for showing key ideas and definitions
- User defines a set of propositional symbols, like P and Q.
- User defines the **semantics** of each propositional symbol:
  - P means “It is hot”
  - Q means “It is humid”
  - R means “It is raining”
- A sentence (well formed formula) is defined as follows:
  - A symbol is a sentence
  - If S is a sentence, then $\neg$S is a sentence
  - If S is a sentence, then (S) is a sentence
  - If S and T are sentences, then $(S \vee T)$, $(S \wedge T)$, $(S \rightarrow T)$, and $(S \leftrightarrow T)$ are sentences
  - A sentence results from a finite number of applications of the above rules

Some terms

- The meaning or **semantics** of a sentence determines its **interpretation**.
- Given the truth values of all symbols in a sentence, it can be "evaluated" to determine its **truth value** (True or False).
- A **model** for a KB is a "possible world" (assignment of truth values to propositional symbols) in which each sentence in the KB is True.

More terms

- A **valid sentence** or **tautology** is a sentence that is True under all interpretations, no matter what the world is actuallylike or how the semantics are defined. Example: “It’s raining or it’s not raining.”
- An **inconsistent sentence** or **contradictio**n is a sentence that is False under all interpretations. The world is never like what it describes, as in “It’s raining and it’s not raining.”
- **P entails Q**, written P |= Q, means that whenever P is True, so is Q. In other words, all models of P are also models of Q.

Truth tables

And

p	q	$p \cdot q$
T	T	T
T	F	F
F	T	F
F	F	F

Or

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

If . . . then

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

Not

p	$\sim p$
T	F
F	T

Truth tables II

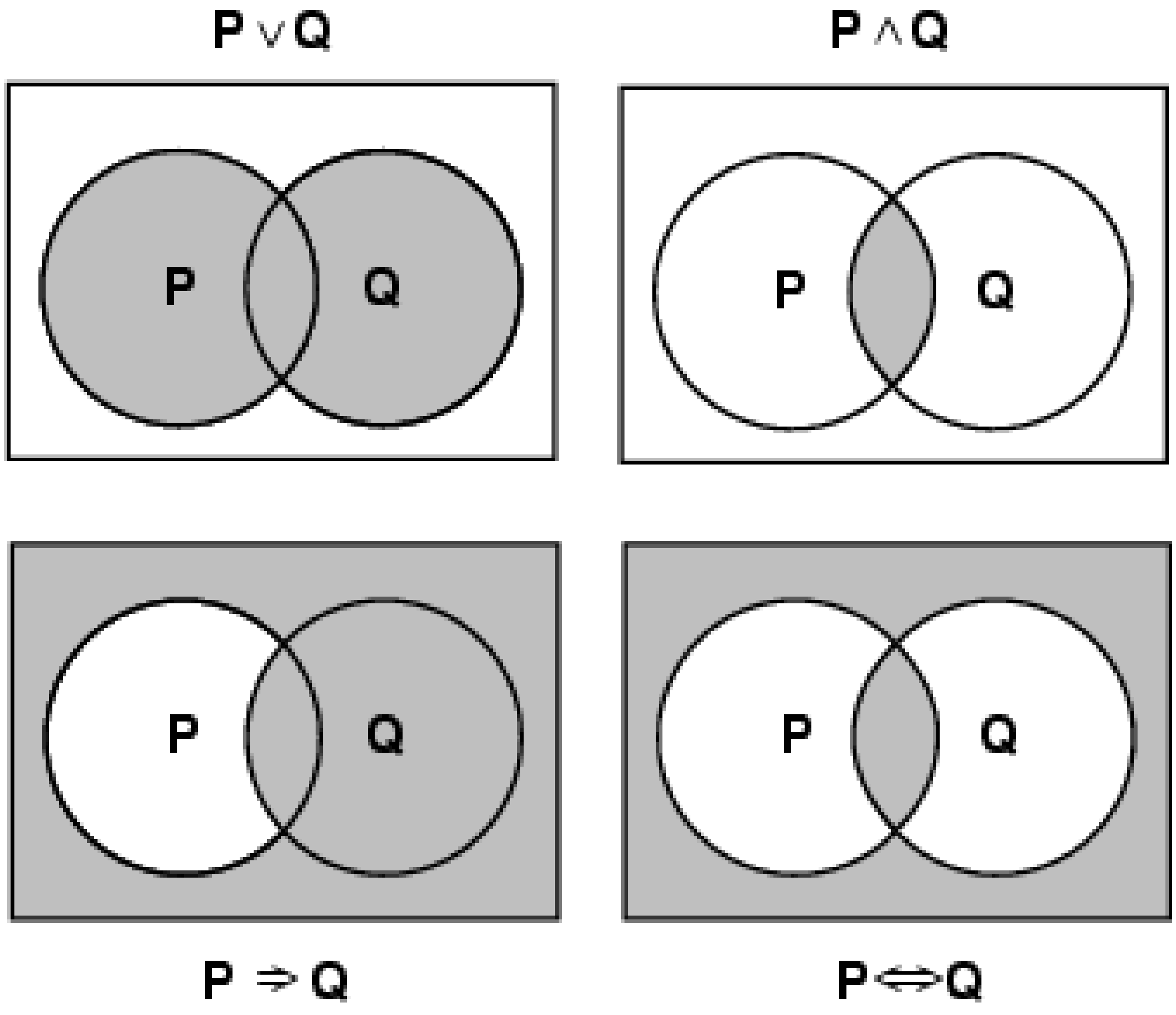
The five logical connectives:

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
False	*False*	*True*	*False*	*False*	*True*	*True*
False	*True*	*True*	*False*	*True*	*True*	*False*
True	*False*	*False*	*False*	*True*	*False*	*False*
True	*True*	*False*	*True*	*True*	*True*	*True*

A complex sentence:

P	H	$P \vee H$	$(P \vee H) \wedge \neg H$	$((P \vee H) \wedge \neg H) \Rightarrow P$
False	*False*	*False*	*False*	*True*
False	*True*	*True*	*False*	*True*
True	*False*	*True*	*True*	*True*
True	*True*	*True*	*False*	*True*

Models of complex sentences



Propositional logic is a weak language

- No notion of objects
- Hard to identify "individuals" (e.g., Mary, 3)
- No notion of relations among objects
- Can't directly talk about properties of individuals or relations between individuals (e.g., "Bill is tall")
- Generalizations, patterns, regularities can't easily be represented (e.g., "all triangles have 3 sides")
- First-Order Logic (abbreviated FOL or FOPC) is expressive enough to concisely represent this kind of information

  FOL adds relations, variables, and quantifiers, e.g.,

  - *"Every elephant is gray":* $\forall$ x (elephant(x) $\rightarrow$ gray(x))
  - *"There is a white alligator":* $\exists$ x (alligator(X) ^ white(X))

First-Order Logic

Outline

- First-order logic
  - Properties, relations, functions, quantifiers, …
  - Terms, sentences, axioms, theories, proofs, …
- Extensions to first-order logic

First-order logic

- First-order logic (FOL) models the world in terms of
  - **Objects,** which are things with individual identities
  - **Properties** of objects that distinguish them from other objects
  - **Relations** that hold among sets of objects
  - **Functions,** which are a subset of relations where there is only one "value" for any given "input"
- Examples:
  - Objects: Students, lectures, companies, cars ...
  - Relations: Brother-of, bigger-than, outside, part-of, has-color, occurs-after, owns, visits, precedes, ...
  - Properties: blue, oval, even, large, ...
  - Functions: father-of, best-friend, Square-root, one-more-than ...

User provides

- **Constant symbols,** which represent individuals in the world
  - Mary
  - 3
  - Green
- **Function symbols,** which map individuals to individuals
  - father-of(Mary) = John
  - color-of(Sky) = Blue
- **Predicate symbols,** which map individuals to truth values
  - greater(5,3)
  - green(Grass)
  - color(Grass, Green)

FOL Provides

- **Variable symbols**
  - E.g., x, y, foo
- **Connectives**
  - Same as in PL: not ($\neg$), and ($\wedge$), or ($\vee$), implies ($\rightarrow$), if and only if (biconditional $\leftrightarrow$)
- **Quantifiers**
  - Universal $\forall$**x** or **(Ax)**
  - Existential $\exists$**x** or **(Ex)**

Sentences are built from terms and atoms

- A **term** (denoting a real-world individual) is a constant symbol, a variable symbol, or an n-place function of n terms.
  - x and $f(x_1, ..., x_n)$ are terms, where each x_i is a term.
  - A term with no variables is a **ground term**
- An **atomic sentence** (which has value true or false) is an n-place predicate of n terms
- A **complex sentence** is formed from atomic sentences connected by the logical connectives:
  - $\neg P$, $P \lor Q$, $P \land Q$, $P \rightarrow Q$, $P \leftrightarrow Q$ where P and Q are sentences
- A **quantified sentence** adds quantifiers $\forall$ and $\exists$
- A **well-formed formula (wff)** is a sentence containing no "free" variables. That is, all variables are "bound" by universal or existential quantifiers.
  - $(\forall x)P(x,y)$ has x bound as a universally quantified variable, but y is free.

Quantifiers

- **Universal quantification**
  - $(\forall x)P(x)$ means that P holds for **all** values of x in the domain associated with that variable
  - E.g., $(\forall x)$ dolphin(x) $\rightarrow$ mammal(x)
- **Existential quantification**
  - $(\exists x)P(x)$ means that P holds for **some** value of x in the domain associated with that variable
  - E.g., $(\exists x)$ mammal(x) $\wedge$ lays-eggs(x)
  - Permits one to make a statement about some object without naming it

Quantifiers

- Universal quantifiers are often used with "implies" to form "rules":

  $(\forall x)\ student(x) \rightarrow smart(x)$ means "All students are smart"

- Universal quantification is *rarely* used to make blanket statements about every individual in the world:

  $(\forall x) student(x) \wedge smart(x)$ means "Everyone in the world is a student and is smart"

- Existential quantifiers are usually used with "and" to specify a list of properties about an individual:

  $(\exists x)\ student(x) \wedge smart(x)$ means "There is a student who is smart"

- A common mistake is to represent this English sentence as the FOL sentence:

  $(\exists x)\ student(x) \rightarrow smart(x)$

  – But what happens when there is a person who is *not* a student?

Quantifier Scope

- Switching the order of universal quantifiers *does not* change the meaning:
  - $(\forall x)(\forall y)P(x,y) \leftrightarrow (\forall y)(\forall x)\ P(x,y)$
- Similarly, you can switch the order of existential quantifiers:
  - $(\exists x)(\exists y)P(x,y) \leftrightarrow (\exists y)(\exists x)\ P(x,y)$
- Switching the order of universals and existentials *does* change meaning:
  - Everyone likes someone: $(\forall x)(\exists y)\ likes(x,y)$
  - Someone is liked by everyone: $(\exists y)(\forall x)\ likes(x,y)$
  - There is a person who loves everyone in the world: $\exists x\ \forall y\ Loves(x,y)$
  - Everyone in the world is loved by at least one person: $\forall y\ \exists x\ Loves(x,y)$

Universal quantification

- $\forall$<*variables*> <*sentence*>

Everyone at VIT-AP is smart:
$\forall x\ At(x,VIT\text{-}AP) \Rightarrow Smart(x)$

- $\forall x\ P$ is true in a model *m* iff *P* is true with *x* being each possible object in the model

- Roughly speaking, equivalent to the conjunction of instantiations of *P*
-

$$
\begin{array}{ll}
 & At(KingJohn, VIT\text{-}AP) \Rightarrow Smart(KingJohn) \\
\wedge & At(Richard, VIT\text{-}AP) \Rightarrow Smart(Richard) \\
\wedge & At(VIT\text{-}AP, VIT\text{-}AP) \Rightarrow Smart(VIT\text{-}AP) \\
\wedge \ldots &
\end{array}
$$

A common mistake to avoid

- Typically, $\Rightarrow$ is the main connective with $\forall$
- Common mistake: using $\wedge$ as the main connective with $\forall$:

  $\forall x\ At(x, VIT\text{-}AP) \wedge Smart(x)$

  means "Everyone is at VIT-AP and everyone is smart"

Existential quantification

- $\exists$<*variables*> <*sentence*>

- Someone at VIT-AP is smart:
- $\exists x$ At(x, VIT-AP) $\wedge$ Smart(x)

- $\exists x\ P$ is true in a model m iff P is true with x being some possible object in the model

- Roughly speaking, equivalent to the disjunction of instantiations of P
-

$$\begin{aligned}
&\quad \text{At(KingJohn, VIT-AP)} \wedge \text{Smart(KingJohn)} \\
&\vee \text{At(Richard, VIT-AP)} \wedge \text{Smart(Richard)} \\
&\vee \text{At(VIT-AP, VIT-AP)} \wedge \text{Smart(VIT-AP)} \\
&\vee \ldots
\end{aligned}$$

Another common mistake to avoid

- Typically, $\wedge$ is the main connective with $\exists$
- Common mistake: using $\Rightarrow$ as the main connective with $\exists$:
-

$$\exists x\ \mathrm{At}(x, \text{VIT-AP}) \Rightarrow \mathrm{Smart}(x)$$

is true if there is anyone who is not at VIT-AP!

Connections between All and Exists

We can relate sentences involving $\forall$ and $\exists$ using De Morgan's laws:

$$(\forall x)\ \neg P(x) \leftrightarrow \neg(\exists x)\ P(x)$$

$$\neg(\forall x)\ P \leftrightarrow (\exists x)\ \neg P(x)$$

$$(\forall x)\ P(x) \leftrightarrow \neg\ (\exists x)\ \neg P(x)$$

$$(\exists x)\ P(x) \leftrightarrow \neg(\forall x)\ \neg P(x)$$

Quantified inference rules

- Universal instantiation
  - $\forall x\ P(x)\ \therefore\ P(A)$
- Universal generalization
  - $P(A) \wedge P(B) \ldots\ \therefore\ \forall x\ P(x)$
- Existential instantiation
  - $\exists x\ P(x)\ \therefore P(F)$ $\leftarrow$ **skolem constant F**
- Existential generalization
  - $P(A)\ \therefore\ \exists x\ P(x)$

Translating English to FOL

Every gardener likes the sun.

$\forall x \; gardener(x) \rightarrow likes(x, Sun)$

You can fool some of the people all of the time.

$\exists x \; \forall t \;\; person(x) \wedge time(t) \rightarrow can\text{-}fool(x,t)$

You can fool all of the people some of the time.

$\forall x \; \exists t \; (person(x) \rightarrow time(t) \wedge can\text{-}fool(x,t))$

$\forall x \; (person(x) \rightarrow \exists t \; (time(t) \wedge can\text{-}fool(x,t)))$

Equivalent

All purple mushrooms are poisonous.

$\forall x \; (mushroom(x) \wedge purple(x)) \rightarrow poisonous(x)$

No purple mushroom is poisonous.

$\neg \exists x \; purple(x) \wedge mushroom(x) \wedge poisonous(x)$

$\forall x \;\; (mushroom(x) \wedge purple(x)) \rightarrow \neg poisonous(x)$

Equivalent

There are exactly two purple mushrooms.

$\exists x \; \exists y \; mushroom(x) \wedge purple(x) \wedge mushroom(y) \wedge purple(y) \wedge \neg(x=y) \wedge \forall z \; (mushroom(z) \wedge purple(z)) \rightarrow ((x=z) \vee (y=z))$

Clinton is not tall.

$\neg tall(Clinton)$

X is above Y iff X is on directly on top of Y or there is a pile of one or more other objects directly on top of one another starting with X and ending with Y.

$\forall x \; \forall y \; above(x,y) \leftrightarrow (on(x,y) \vee \; \exists z \; (on(x,z) \wedge above(z,y)))$

More Examples

- Everyone who loves all animals is loved by someone
- Anyone who kills an animal is loved by no one
- Jack loves all animals.

$$\forall x\, [\forall y\, Animal(y) \Rightarrow Loves(x, y)] \Rightarrow [\exists y\, Loves(y, x)]$$

$$\forall x\, [\exists y\, Animal(y) \wedge Kills(x,y)] \Rightarrow [\forall z\, \neg Loves(z,x)]$$

$$\forall x\, Animal(x) \Rightarrow Loves(Jack, x)$$

More Examples

- "John has at least two umbrellas"

there exists x: (there exists y: (Has(John, x) AND IsUmbrella(x) AND Has(John, y) AND IsUmbrella(y) AND NOT(x=y))

- "John has at most two umbrellas"

for all x, y, z: ((Has(John, x) AND IsUmbrella(x) AND Has(John, y) AND IsUmbrella(y) AND Has(John, z) AND IsUmbrella(z)) => (x=y OR x=z OR y=z))

More Examples

- “Duke’s basketball team defeats any other basketball team”

for all x: ((IsBasketballTeam(x) AND NOT(x=BasketballTeamOf(Duke))) => Defeats(BasketballTeamOf(Duke), x))

- “Every team defeats some other team”

for all x: (IsTeam(x) => (there exists y: (IsTeam(y) AND NOT(x=y) AND Defeats(x,y))))

CLAUSAL FORM

(1) LITERAL: A literal is either an atomic sentence or negation of an atomic sentence.

E.g., $P(a)$, $\neg P(b)$

(2) A clausal sentence is either a literal or a disjunction of literals

e.g., $P(a)$, $\neg P(b)$, $P(a) \vee \neg P(b)$

(3) A clause is a set of literals

$\{P(a)\}$, $\{\neg P(b)\}$, $\{P(a), \neg P(b)\}$

Step 1: Remove Implications

$$P \rightarrow Q \equiv \neg P \vee Q$$

$$P \leftrightarrow Q \equiv (\neg P \vee Q) \wedge (P \vee \neg Q)$$

Step 2: (Negations In)

$$\neg\neg P \equiv P$$

$$\neg(P \wedge Q) \equiv \neg P \vee \neg Q$$

$$\neg(P \vee Q) \equiv \neg P \wedge \neg Q$$

$$\neg \forall x P(x) \equiv \exists x \neg P(x)$$

$$\neg \exists x P(x) \equiv \forall x \neg P(x)$$

Step 3: Standardize variables

$$\forall x P(x) \vee \forall x Q(x) \longrightarrow \forall(x) P(x) \vee \forall y Q(y)$$

Step 4: $\exists x\, P(x) \equiv P(a)$ [Existentials Out]

$$\forall x\,(P(x) \wedge \exists z\, Q(x,y,z)) \equiv \forall x\,(P(x) \wedge Q(x,y,f(x,y)))$$

Step 5: $\forall x\,(P(x) \wedge Q(x,y,f(x,y))) \equiv P(x) \wedge Q(x,y,f(x,y))$ [Alls Out]

Step 6:

$$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$$

$$(P \wedge Q) \vee R \equiv (P \vee R) \wedge (Q \vee R)$$

$$P \vee (P_1 \vee \ldots \vee P_n) \equiv (P \vee P_1 \vee \ldots P_n)$$

$$(P_1 \vee \ldots \vee P_n) \vee P \equiv (P_1 \vee P_2 \vee \ldots \vee P_n \vee P)$$

$$P \wedge (P_1 \wedge \ldots \wedge P_n) \equiv (P \wedge P_1 \wedge \ldots \wedge P_n)$$

$$(P_1 \wedge \ldots \wedge P_n) \wedge P \equiv (P_1 \wedge P_2 \ldots \wedge P_n \wedge P)$$

Distribution.

Step 7: $P_1 \wedge \ldots \wedge P_n \Longrightarrow \begin{array}{c} P_1 \\ \ldots \\ P_n \end{array}$

$$P_1 \vee \ldots \vee P_n \equiv \{P_1, \ldots, P_n\}$$

Operators Out.

Example 1:

$\exists y (g(y) \wedge \forall z (r(z) \rightarrow f(y,z)))$

I $\exists y (g(y) \wedge \forall z (\neg r(z) \vee f(y,z)))$ — Step1 (Implication, I)

N $\exists y (g(y) \wedge \forall z (\neg (r(z) \vee f(y,z)))$ — Step2 (Negations In, N)

S $\exists y (g(y) \wedge \forall z (\neg r(z) \vee f(y,z)))$ — Step3 (Standardize, S)

E $g(greg) \wedge \forall z (\neg r(z) \vee f(greg, z)))$ — Step4 (Existentials Out, E)

A $g(greg) \wedge (\neg (r(z) \vee f(greg, z)))$ — Step5 (Alls out, A)

D $g(greg) \wedge (\neg (r(z) \vee f(grg, z)))$ — Step6 (Distribution, D)

O $\{g(greg)\}$

$\{\neg r(z), f(greg, z)))$

— Step7 (Operators Out, O)

Example

$$\forall y\,(\neg g(y) \vee \exists z\,(r(z) \wedge \neg f(y,z)))$$

E $\quad \forall y\,(\neg g(y) \vee (r(h(y)) \wedge \neg f(y, h(y))))$

A $\quad \neg g(y) \vee (r(h(y)) \wedge \neg f(y, h(y)))$

D $\quad (\neg g(y) \vee r(h(y))) \wedge (\neg g(y) \vee \neg f(y, h(y)))$

O $\quad \{\neg g(y),\ r(h(y))\}$

$\quad\quad \{\neg g(y),\ \neg f(y, h(y))\}$